GET A JOB! Special Career Supplement Inside!

3D
Design

Characters
ALIVE!

- Hash Animation:Master 99
- Softimage|3D
- Credo Life Forms

Modeling in
auto•des•sys form•Z

BIG KAHUNA
Nominees Revealed!

JUNE 1999
Vol. 5, No. 6
U.S. $3.95 Canada $4.95
www.3d-design.com
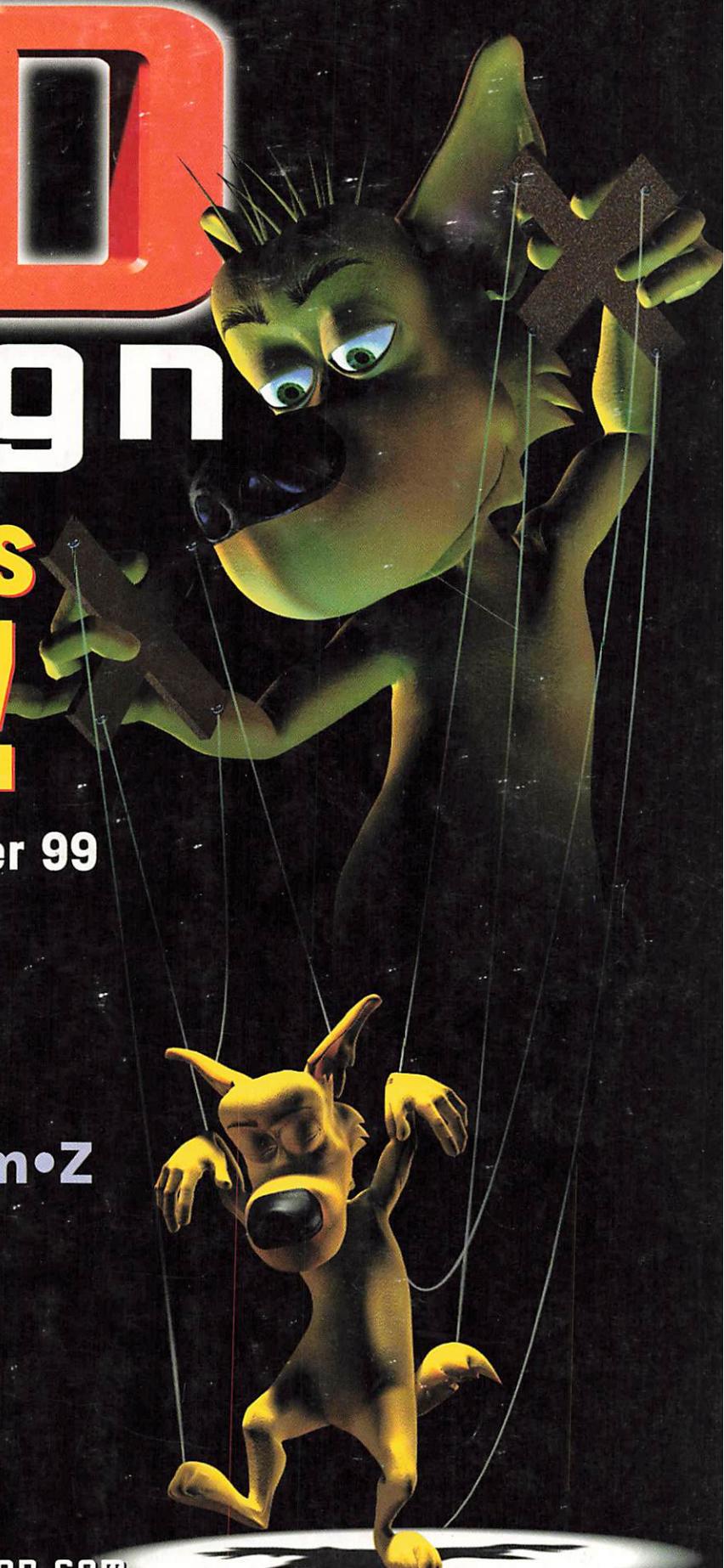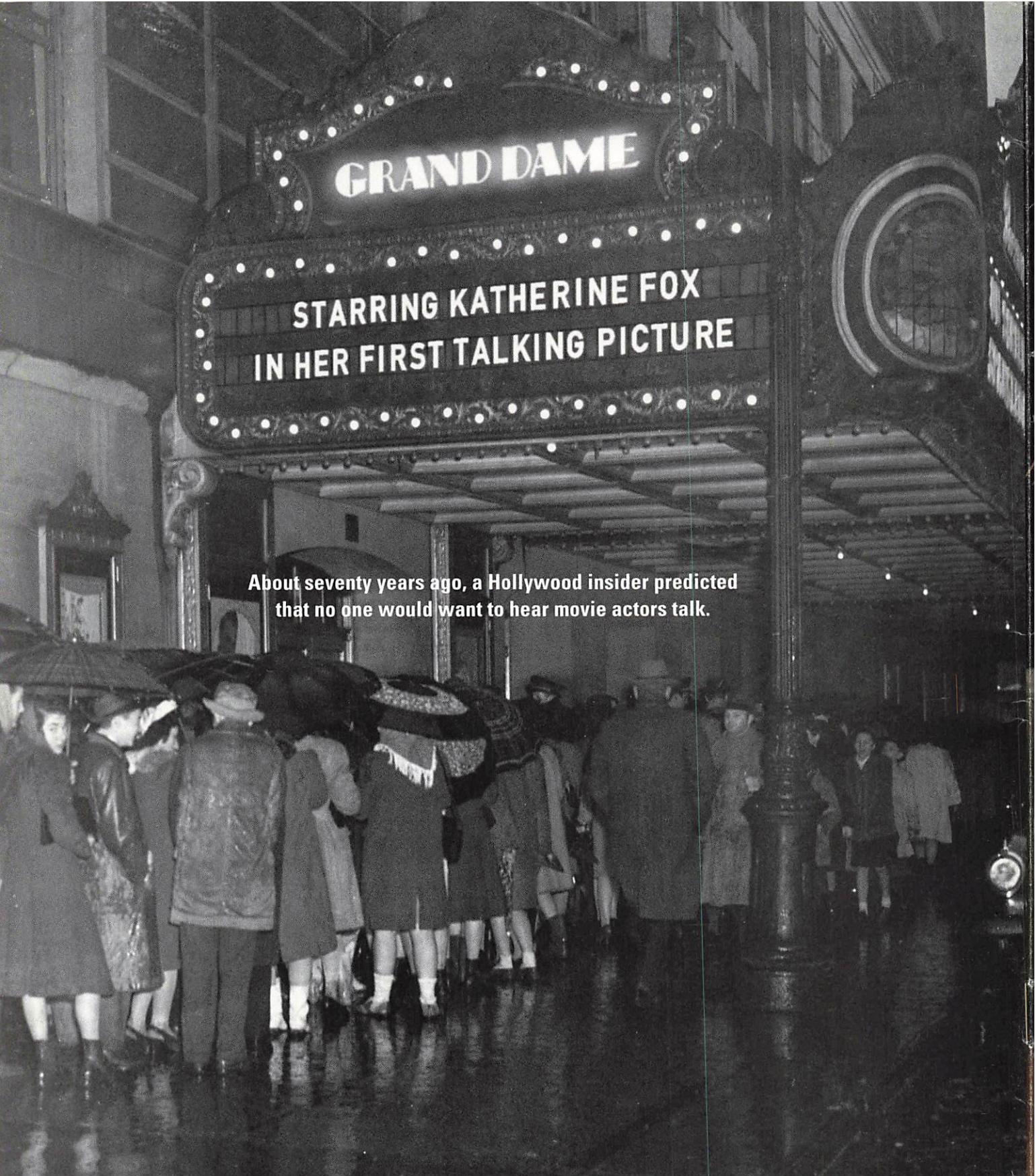
GRAND DAME

STARRING KATHERINE FOX
IN HER FIRST TALKING PICTURE

About seventy years ago, a Hollywood insider predicted
that no one would want to hear movie actors talk.

Hop on an iMac™ and in just ten minutes you could be surfing the internet and e-mailing everyone
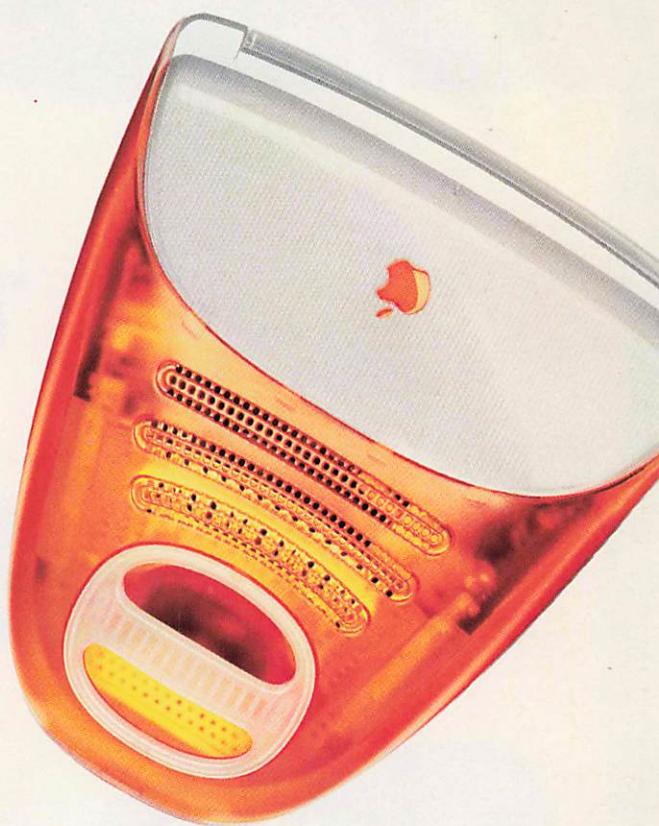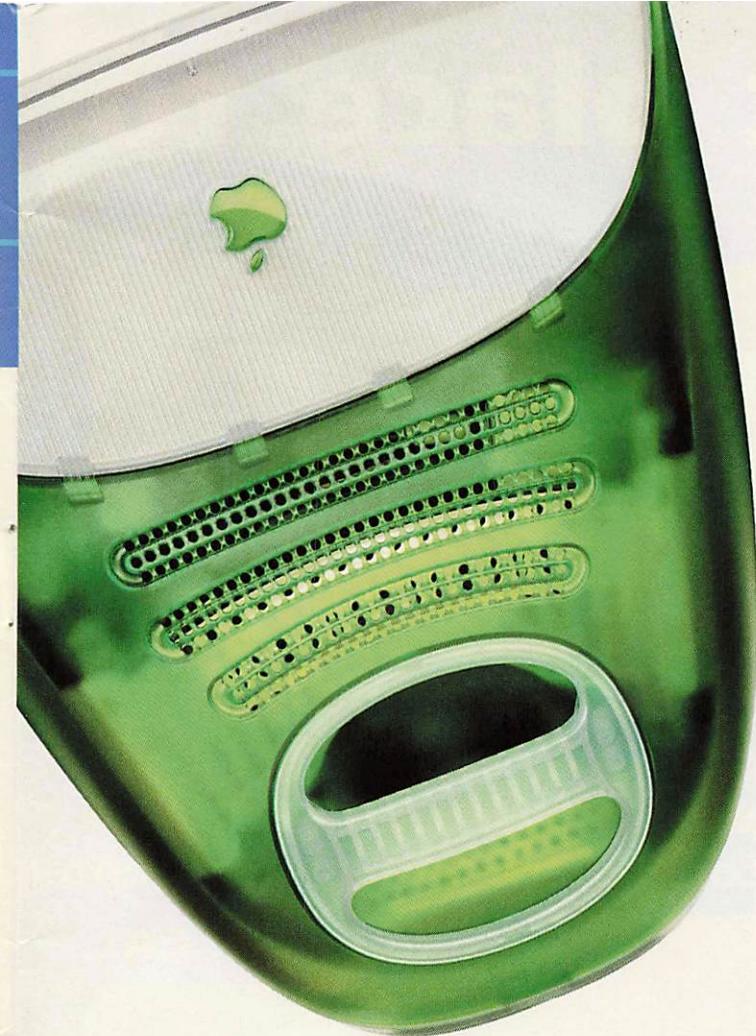
The thrill of surfing.
The agony of choosing a color.

and their brother. Now for the hard part: what color will it be? www.apple.com 🍎 Think different.

learn new skills and see new gear.

Mark your calendars for four intense days of in-depth training on digital video technology and business. At the **DIGITAL VIDEO CONFERENCE** you'll learn from top professionals in the field how to leverage the latest technology and optimize your work for the full range of delivery options.

COURSE TOPICS INCLUDE:

2D and 3D video compositing
Optimizing video for broadband and streaming on the web
Optimizing video for CD-ROM and DVD
QuickTime 4.0
Video capture & non-linear editing
Advanced layering in After Effects
and much more!

Request a catalog today!
1-800-789-2223 / www.dvexpo.com
outside the U.S. call 817-255-8050

Yes! Send me a free copy of the Digital Video Conference & Exposition Catalog!

Name

Title

Company Address

| **City** | **State** | **Zip** |

| **Phone** | **Fax** | **Email** |

Fax to: 888-811-7391 · *outside the U.S. fax to 817-255-8070*

Or visit our web site and sign up on the mailing list! www.dvexpo.com

Priority Code: DGJ0699

3D Design

"You Wanna Iguana?" by Kim Oravecz, 1999 Big Kahuna Nominee.
Created with LightWave 5, Lparser, Bryce 2, Photoshop 4, Detailer, and Forge.

June

OUT OF MY M!ND

3D at the Precipice

My descent into the inner sanctum of 3D began in a flurry of secret meetings with publishers, conference organizers, webmasters, special-ops personnel, and shadowy upper-level management. The Corporation, it seemed, had identified 3D as the target growth opportunity of the new millennium, and they had tapped my shoulder with a terse invitation to climb aboard as editor-in-chief of their flagship publication and battering ram, *3D Design* magazine.

Filling Kelly Dove's ruby slippers was a tall order. Even taller was the demand to serve a readership whose accomplishments in 3D art and animation are literally awesome, and whose demand for the most accurate, relevant, and in-depth information might be characterized as rabid. I swallowed hard and accepted the assignment.

Little did I know that 3D technology, after a period of retrenchment during 1998, lately has been marching toward the edge of a precipice that divides the old world of two dimensions from a virtual universe on every desktop. A week after I joined *3D Design*, AT&T finalized its acquisition of cable giant TCI amid talk of new data services fast enough to do Internet-based 3D right. A few days later, Sony released the details about the forthcoming PlayStation 2, a gaming supercomputer capable of pushing 66,000,000 polygons per second, apparently trumping Sega's 128-bit Dreamcast.

The PlayStation 2 will deliver 3D content on DVD. Online, the format might be VRML—or rather its new XML-enhanced incarnation X3D or a proprietary technology such as NTT Software's InterSpace or Cycore's Cult3D. The former mingles real-time audio and video chat within low-bandwidth 3D worlds; you can see it on the *3D Design* web site (www.3d-design.com). As for Cult3D, Cycore reps dropped by to demonstrate new interactive features. They showed off a 3D mock-up of a PDA/cel phone that not only rotated in response to mouse clicks but even sent e-mail messages clicked on its keyboard— a fully functional virtual demo.

These forces will drive 3D deep into the mainstream, but the Trojan Horse has yet to open: Microsoft's GDI 2000. Scheduled to debut at WinHEC in April, GDI 2000 is an underlying technology expected to give Windows a full-fledged 3D GUI.

Not only will the floodgates open for 3D content, the tools for creating content are due to improve dramatically. An amazing number of vendors have released or promised substantial upgrades this year: Kinetix 3D Studio MAX 3, Alias|Wavefront Maya 2, Nichimen Mirai, NewTek LightWave 6, MetaCreations Bryce 4, Side Effects Houdini 2, not to mention the near-light-speed Intergraph Wildcat 4100 board. What a playground! And here I am, smack dab in the middle of it.

I promise I won't let the opportunity go to waste. 3D technology is rocketing into a new era of creative power and ubiquitous delivery, and *3D Design* is committed to bringing artists and animators the information they need to make the most of it. Look for reviews that slice through the hype, case studies of innovative 3D projects, technical articles, and step-by-step tutorials that will help you hone your edge and realize your vision.

In coming months, we'll sharpen our angles to make sure they cut straight to the heart of your art. If there's anything you've always wanted to see in *3D Design*, now is the time to let me know. It's shaping up to be a wild ride, and I want to make sure you come along with us. ●

Ted Greenwald

Ted Greenwald, editor-in-chief

FEEDBACK

Not at all avid

► **The article** "For the Avid Editor" (April 1999, p. 19) is so full of inaccuracies that it appears Dan Ablan either did not spend much time with the program or was under pressure from you and Avid to give a glowing recommendation.

The most blatant fabrication is, "Macintosh users of MCXpress are working with version 2.0, which is similar to the NT 1.6 version, according to Avid." First, the Mac MCXpress was discontinued in April 1998. Second, after nearly two years of implying the two products had some parity, Avid changed its story and stated the two were *never* intended to be alike in any way; MCXpress Mac was for professional media content creators and video editors, MCXpress NT was not.

The "new features" listed for NT 1.6 have been a huge source of irritation. Some were advertised as being part of the product since revision 1.1 but never worked properly. Yet Avid saw fit to charge for an upgrade that users hoped was a bug fix. Two of the main "new features," OMF import/export and EDL import/export, still did not work in 1.6.

Ablan also fails to point out that, even with the Targa 2000 RTX board, MCXpress for NT is still not a real-time system until you add, for double the cost of the software, the Real-Time Option. Also, animators using MCXpress for adding audio tracks should be warned that MCXpress can lose or gain frames while importing and exporting audio.

Especially noteworthy, yet missing from the article, is the fact that MCXpress effects do not support field rendering. If you want to end a clip on a freeze of its last frame, it will drop to half-resolution with line doubling used on a single field of the final frame. In essence, motion effects like freeze-frame cannot be used in a professional environment.

The article should have warned users to steer clear. You have done a disservice to your readers by trumpeting on the cover, "Get Avid MCXpress for NT."
Wayne M. Cole
Infinity Heart Productions

I've used MCXpress for two years on many different projects. Also, in my years writing about 3D and video, I've never felt pressure from any advertiser. If the tone of the article came across as such, I apologize.

At NAB in Las Vegas last year, Avid said Mac MCXpress v. 2.0 was the same product as NT 1.6. This was confirmed by an Avid reseller just before I wrote the article and before I upgraded our system. As it turns out, despite their common name, the two products are quite different, with different components, features, prices, and code bases. In April 1998, the Mac version was not discontinued but renamed XPress. Perhaps Avid should have done a better job keeping their user base informed.

Effects Affected An info box in our Adobe After Effects review (May 1999, p. 43) listed two vendors incorrectly. The Knoll Lens Flare Pack is available from Puffin Designs (www.puffindesigns.com), and Final Effects is available from ICE (www.iced.com).

I was aware of the Real-Time Option and take responsibility for failing to mention it in the article. I agree the real-time features are overpriced and not worth the extra cost.

In two years, I've never had audio problems such as the one you've mentioned. Regarding field rendering, you're absolutely correct, and it's very annoying. For our studio, going from simple animations and playback to fairly decent editing capabilities was a leap at first. Our demands have changed (even since the aricle was written months ago), and now the MCXpress has been collecting some dust. Also, now that Adobe's Premiere 5.1 supports DPS boards, much of our finishing has been done in Premiere. We've also started using in:sync Speed Razor. Overall, I'm not extremely happy with MCXpress, and I thought the article had a "so-so" tone. Still, I feel it could be a worthwhile investment for NT users who need more editing capbabilities than a simple DV solution. **—Dan Ablan**

Regarding our April cover line about MCXpress, we meant the words "Get Avid" to be read as "get excited, get eager," the literal meaning of the word "avid." We missed the obvious interpretation, "buy this product." It was a poor choice of words, but not meant as an endorsement of MCXpress. **—The Editors**

Real grunge

► **I've been reading** *3D Design* for a little over a year now, and I wait by my mailbox for every new issue. But this month's article "Stealing Reality" (May 1999, p.21) —wow! Alex Lindsay is a tremendous writer and has opened a new nerve in my cranial compartment. Grunge maps? Never heard of 'em before now, but they will become a regular part of my 3D work from now on.
Donald McDade
Syracuse, New York ●

Of course it leaves you speechless. It's designed for VIS

silicon graphics 320

sgi

Visualize, Win a Prize

● **Graphisoft, creators of the popular ArchiCAD architectural CAD software, recently announced a diverse collection of themes for its sixth annual Graphisoft Prize international design competition.**

Bluebeard's Castle by Gordon Rossol, second place winner in last year's Graphisoft Prize competition.

The contest invites architectural professionals and students to interpret buildings that exist in their imaginations. Every year, a collection of themes is drawn from literature, film, poetry, mythology, music, and fine art. This year's categories are:

- Blue Sky Mine, from the song "Blue Sky Mining" by Midnight Oil.
- The Castle, from Franz Kafka's novel The Castle.
- The Cheap Hotel, from William Gibson's sci-fi novel Neuromancer.
- House of Fame and Hall of Rumor, from Chaucer's poem The House of Fame.
- Land of Extreme Felicity in the West, from Chinese mythology and the novel Hsi Yu Chi (Travels in the West).
- Maggie's Farm from the song of the same name by Bob Dylan.
- The Diner, from Edward Hopper's painting Nighthawks.
- The Palace of Haroun al-Rachid, from Tales of 1001 Nights.
- The Pharaoh's Palace, from Thomas Mann's Joseph and His Brothers.

- The Planet of the Businessmen, from Antoine de Saint Exupery's Little Prince.
- Rick's American Café, from the film Casablanca directed by Michael Curtiz.
- The Zone, from the film Stalker directed by Andrei Tarkovsky.

Last year's winning team was Andrew Maynard and Stephen Mees, who impressed judges enough to take the grand prize with an architectural interpretation of Bulgakov's tale The Master and Margarita, which in Mees' words "represents the domain of the devil, paved with the litany of wealthy and influential citizens and intelligentsia, right up to the door of the madhouse."

It's free to enter the contest; participants must, however, register at www.gsprize.com. Separate award categories have been established for students and professionals, and student entrants must be at the university level of architecture, landscape architecture, urban planning, or a related field. Find out more online at www.graphisoft.com.
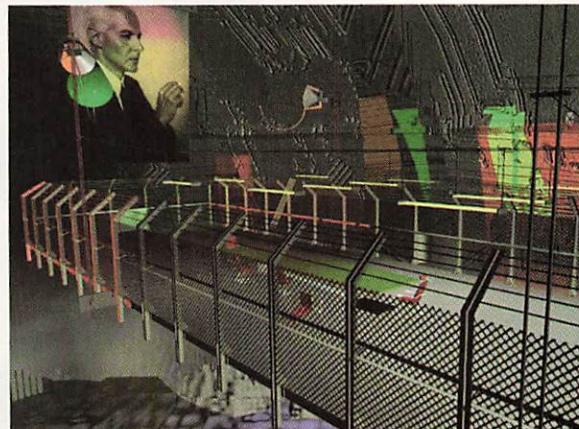
Oscar's Not a Grouch

Proving once again that computer graphics have infiltrated the mainstream, Blue Sky|VIFX Studios was the deserving recipient of the Academy Award for Best Animated Short Film for its seven-minute piece, Bunny. The project used proprietary lighting and rendering techniques to simulate radiosity. This gave the characters and sets a richness and visual depth not seen before in computer animation and raised the bar for realism in 3D animation.

The other CG winners in the Oscar lineup were the teams behind What Dreams May Come, which took the statuette for Best Visual Effects. The film's effects were created by a slew of studios, including Mass.Illusions (subsequently Manex Visual Effects), Pacific Ocean Post, Digital Domain, Giant Killer Robots, Shadow Caster, and Lunarfish. Together they created visually stunning sequences, including Robin Williams and

A contest for CAD dreamers, the debut of Pentium III Xeon, the growing marketshare of AMD, the scoop on Kinetix and Discreet, and the next wave in home gaming machines in this month's news.

Rapid 3D #5

Cuba Gooding Jr. interacting with environments that resemble famous paintings. The effects were beautiful and compelling. *3D Design* congratulates all the artists involved on a job well done.

Xeon Rising

A month after the launch of the consumer-oriented Pentium III, Intel in March unveiled the Pentium III Xeon line of processors in San Francisco. Topping out at 550MHz, the new Xeon chips, formerly code-named Tanner, are Intel's fastest to date and come with up to 2MB of L2 backside cache. While these chips herald the imminent arrival of the next generation, Cascade (PIII Xeon with 0.16 micron silicon traces compared to today's 0.25), and ultimately the 64-bit Merced architecture, Intel has committed to reaching speeds of up to 1GHz by the year 2000.

All Pentium IIIs carry the Katmai instruction set of 70 new registers designed primarily to speed up 3D graphics and multimedia.

AMD Gains

AMD has recently enjoyed a market leader position in the consumer PC space by surpassing Intel in retail units sold for the first time ever. This hasn't been enough to keep the red ink from flowing, however; AMD recently posted a record loss and consequently laid off 300 employees. While AMD has been popular in gaming for a while, it has yet to break into the more lucrative high-end PC market, where profit margins

are sweeter than in the sub-$1,000 PC space.

So it comes as no surprise that AMD is going after the professional workstation market in a big way with the recent announcement of the AMD K7 processor. AMD is touting the K7, with its 3DNow architecture, as a "Pentium killer." The preliminary data, although sparse, is impressive. Rumor has it that the K7's clock speed will be faster than the current Pentium III, and if compatibility questions can be proven moot, AMD will give Intel a run for its money. The professional market will become increasingly critical to AMD as processors get less expensive and profits in the consumer space continue to shrink.

Kinetix Gets Discreet

Announced March 16 at the Game Developers Conference in San Jose, Kinetix completed its merger with Montreal-based Discreet Logic. As a change to the initial plan, the whole of Discreet Logic and Kinetix will now be known as Discreet.

Discreet will maintain offices in Montreal for the high-end IRIX-based software group, and the San Francisco division will be responsible for 3D Studio MAX, Character Studio, paint*, effect*, edit*, and Lightscape (formerly known as light*, which was formerly known as Lightscape).

Through all the merger mania, Discreet has continued to pursue its commitment to the 3D market by announcing 3D Studio MAX 3, a substantial
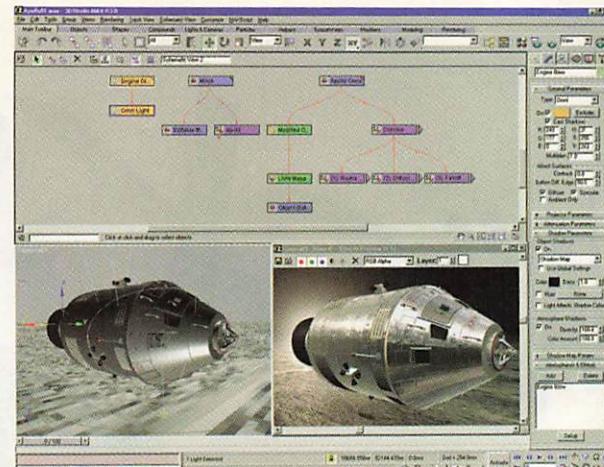
Here's a look at 3D Studio MAX 3.0's customizable interface.

upgrade to an already popular 3D product. If MAX 2 constituted many small dabs of an "upgrade brush" onto the software canvas, MAX 3 goes at it with a paint roller.

The MAX 3 user interface is totally customizable, and UI features found in other popular apps can now be enjoyed by MAX users. The other major griping point, the rendering engine, has had a complete overhaul, and now delivers much higher quality images. New shaders, project workflow enhancements, scene and object referencing, and more improvements make this a solid upgrade from a company that has been anything but discreet in bringing powerful software to the 3D community.

Serious Play

Sony recently announced the specs on its long-anticipated PlayStation 2, the successor to the most popular gaming machine yet, the original PlayStation. Based on a 128-bit Emotion Engine CPU running at 300MHz, the unit claims to push 10 to 15 million textured, lit, and shaded polys per second. It will be backwards compatible with games made for its predecessor.

Sony also announced that Softimage's development group will help create 3D animation tools for developers working on PlayStation 2 games.

The machine, which is based on a DVD player, will not play DVD videos right out of the box, but the capability to add support for DVD video is not out of the question.

Meanwhile, Sega is ramping up for the U.S. launch of its latest game platform, Dreamcast, which is already shipping in Japan. Based on the Hitachi SH4 RISC CPU at 200MHz and using the second-generation PowerVR chipset from NEC, Dreamcast runs a custom SegaOS and Windows CE. Estimates for polys per second are three to four million. Additionally, the unit has 16MB of main memory, 8MB of video memory, and 2MB of sound memory.

Dreamcast's reception has been good but not overwhelming. Sega is relying on this system to pull it out of the sales funk it's experienced the last few years. Dreamcast is expected to make its U.S. premiere in September 1999. ●

Chris Tome is technical editor for *3D Design*. You can e-mail him at ctome@mfi.com.

Prosperous Future

▶ **The next generation of** Nichimen's N-World suite of real-time 3D content creation tools is called Mirai. Created for game developers and high-end character animators, Mirai features subdivision surface modeling, 2D and 3D paint, skeletal modeling, inverse kinematics, bio-mechanical motion editing, a channel-based, non-linear motion editing system, particles and physical simulation, and photorealistic rendering. Also, Mirai offers support for Game Exchange 2 standards for ASCII data export and conversion. Designed to resemble a fully functional 3D operating system, Mirai makes it possible to control object visibility and sensitivity on a per-editor basis, enabling users to work on different objects in multiple editors simultaneously. Mirai is priced at $6,495 and comes with complete online documentation, a texture library, and a motion capture library from House of Moves. Contact Nichimen Graphics at 310-577-0500. (www.nichimen.com)

RAPID 3D NO. 151

MAX Textures

▶ **Platinum Picture's Texture Layers** is a plug-in for 3D Studio MAX that allows complex mapping composition over a 3D model using layered UVW coordinates. Texture Layers supports up to 1,000 mapping channels per object, allowing users the freedom to assign multiple planar, spherical, cylindrical, and shrink wrap mapping in any of the layers. Visit Platinum Pictures web site for beta purchase information, or call 608-355-1097. (www.platinumpictures.com)

RAPID 3D NO. 152

True to Life

▶ **trueSpace4 users** can take advantage of a compilation of third-party plug-ins and custom shaders via the pluSpack2 from Caligari Corp. This compilation offers Positron's TrU-V to create a single, custom bitmap for polygonal objects based on the objects unique geometry; Quantum Impulse's SpaceTime Morph, a targeting morph system; ThermoClay2, a smoothing plug-in; and MotionClay, which allows users to animate low-polygon control meshes using vertex animation or bones animation, then smoothly subdivides the control mesh just prior to render time. Also included is Frank Ramsay's Crumple Shader, a displacement shader designed to create animated ocean waves, crumpled paper, or rock walls, and the Perlin Noise Shader based on the noise algorithm developed by Ken Perlin. Urbank Velkavrh's post-processing Intrude-R Shader is included, as is windmill fraser's "sphereglo" shader and wipe plug-in. The pluSpack2 is available for $99. Contact Caligari Corp. at 650-390-9600 or 800-351-7620. (www.caligari.com)

RAPID 3D NO. 153

Cult of After Effects

▶ Cult Effects plug-ins from Cycore Computers are designed for Adobe After Effects v. 3.1/4.0 and focus on film post-production. Cult Effects is a mix of 18 effects filters, including advanced fractal-based simulations, camera lens distortions, animated electrical discharges, and color tools and gradients. Using the fractal-based effects, all fractals can be animated with full user control to form anything from caustics and clouds to flowing lava, water, or gas. The animated electrical discharges give the impression of being trapped or bounced inside a sphere or appear to outline an object in its path. Cult Effects is available on the Mac or Windows platform and is priced at $695. Contact Cycore Computers at +4618 65 65 60. (www.cult-effects.com or www.cult3d.com)

RAPID 3D NO. 154

Virtual Sculpture

▶ **Two plug-ins for the Softimage|3D Plus and Softimage|3D Performance packages are available from Konshus Technologies. The Grabber and Alter by Curve** plug-ins are designed to give users control over the modeling process through virtual sculptor's tools and versatile animation effects. These plug-ins allow animators to superimpose their deformation over existing ones. For example, they can be used to create mass and body-fat movements, bend or distort models, and simulate character expressions—all possible without needing the model's hierarchy. Alter by Curve is priced at $429 and Grabber is priced at $359. Contact Konshus Technologies at 450-458-1484. (www.konshus.com)

RAPID 3D NO. 155

Instant Modeling

▶ **InSpeck-3D is a** computer-aided, non-contact optical 3D digitizer from InSpeck inc. that captures the 3D form and texture of a given surface. Options available for InSpeck-3D units include color or black versions, both of which can acquire texture and 3D coordinates of up to 300,000 points in 0.3 seconds. InSpeck-3D uses a halogen white light source instead of a laser, which provides lighting for color image capturing that's harmless to the human body. Captured models can be imported into popular 3D modeling and animation software, including Softimage|3D and 3D Studio MAX. InSpeck-3D is priced between $19,500 and $35,000, depending upon the application, and it can be configured according to the user's specific needs. Contact InSpeck inc. at 514-284-1101. (www.inspeck.com)

RAPID 3D NO. 157

3D Text

▶ **Cool 3D v. 2.5 from** Ulead Systems is a 3D text manipulation program that allows users to create and interact with onscreen projects in real time. Version 2.5 comes bundled with 12 special effects plug-ins, and it adds support for field-based output, which gives users video and high-resolution graphic output and Direct3D acceleration. The plug-ins include effects for frame, imprint, and hollow, which are used for manipulating title borders and edges, and glow, shadow, and motion blur for applying effects to entire projects. Other additions include new ways of assigning motion to titles and individual elements, such as Token-based motion effects, surface and path animation, and dance, which makes objects dance around. Cool 3D v. 2.5 is priced at $49.95. Contact Ulead Systems at 310-523-9393. (www.ulead.com)

RAPID 3D NO. 156

Made in the Shade

▶ **For version 2.0,** SpeedRender from Imagica Corp. has added new mental ray shaders and a custom effect tool called SpeedMetaball, which raytraces metaballs without creating polygons. The SpeedRender 2.0 collection for Softimage|3D has added the following shaders: Ocean, Flame, Streak, and LightMap, which is a 2D texture shader working in combination with SpeedLight to create a light map with MIPmap. Other features include SpeedMaterial, which enhances the transparency by independently setting RGB transparency, and MaterialMixer, which blends different materials with a texture map. SpeedRender 2.0 is compatible with Softimage|3D 3.7 SP1, and pricing starts at $950. Contact Imagica Corp. at 310-306-4180. (www.speedrender.com)

RAPID 3D NO. 158

1 Engineer. 1 Idea.
1165 Parts.
20 Days.
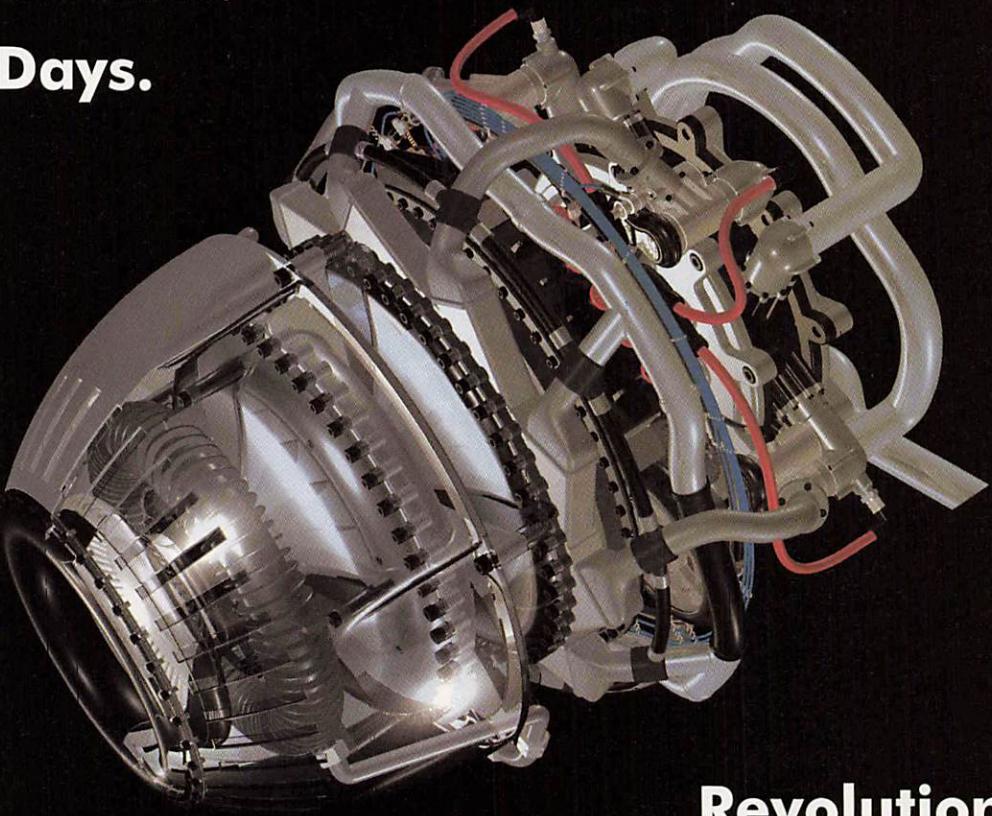
Revolutionary.

For a long time industrial designer Simon Floyd had an idea for a new type of engine. He also had at his fingertips one of the leading "high end" CAD systems, but the one thing he didn't have was spare time. "This is a complex assembly and the task of designing it in any conventional CAD system was just too daunting. However, once I got my hands on IronCAD, I finally had the power to realize my vision," says Mr. Floyd. IronCAD's unique Drag & Drop Solid Modeling and Design Flow™ Architecture gave him the power to not only work fast, but to innovate while he designed. He's become so enthusiastic about IronCAD, he now offers it to his clients. To see a case study of Simon Floyd's awesome engine design and to learn more about how IronCAD is powering the next industrial revolution, log on.

www.**ironcad**.com

800.339.7304

**IRON
CAD**
VISIONARY DESIGN SYSTEMS

Join the revolution or be left behind.

Theme and Variations in form·Z

▲ **Not long ago, as I walked down the street one afternoon, something bright and flashy caught my eye. I found myself staring at a bizarre floor lamp in the window of a trendy junk store. It reminded**

me of the '70s-era lamps known as starlights due to the multiple star-shaped lights floating atop ray-like fixtures. Immediately, as is my tendency upon seeing unusually shaped objects, I began to think about recreating it as a 3D model. But how to approach it?

While the umbrella term "3D modeling" tends to bring to mind the various stages of the modeling process and its often spectacular outcome, most 3D modeling packages can also serve as wonderful tools for exploring and developing design ideas.

A fairly intuitive 3D modeling system such as auto·des·sys form·Z can easily be used as a means to articulate and communicate design intentions through all levels of development. The real power of 3D modeling is that it enables artists to explore many variations and design possibilities starting from the ground up.

In homage to the trend of collecting retro items from the '60s and '70s such as lava lamps and bell bottoms, we're going to design and model a '70s-style starlight lamp and examine alternative forms, materials, and lighting conditions. We'll start with a simple sketch idea and explore its development into form.

The purpose of this tutorial is to experiment with design possibilities in the pro—cess of building a model. I will present the basic approach used in making the lamp, including a sequence of steps and sugges-

tions for design alternatives that could be useful for further explorations along the way. I encourage you to make changes and try variations based on your own design aesthetic.

But first, a note on form·Z versions. The release of form·Z 3.0 earlier this year provides a major upgrade to the capabilities of the software (see, "What's New in form · Z 3.0," p. 32). Although this tutorial is written using v. 3.0, it will work fine with v. 2.9 with a few exceptions noted in the text. One major new feature to be aware of is the Tool Options palette, which allows the immediate selection of options that affect the active tool. The options can also be set, as in previous versions, by double clicking on the tool to open a dialog. Another wonderful new feature that supports the exploration of design alternatives is the ability to modify the parameters of objects created with the Revolve, Sweep, and Skin tools by simply selecting them again with the same tool (or with the Query tool).

The advanced modeling capabilities of form·Z can be used to explore and develop design ideas from the ground up. Here are some tips on how to approach your next form·Z project, along with a look at some of the new features in v. 3.0.

THIRDDIMENSION

Drawing A Sketch Most designers usually begin with quick sketches before tackling complete models. Typically, these sketches are made using traditional media such as fat pencil on paper (heavy lines used to convey intention without too much concern for detail and exact dimension). We're going to try something a little different. Launch form·Z (a new project with a Model window will be created automatically) and open a new Draft window. Within the form·Z Drafting module, make a quick sketch of the lamp using lines, polystreams, and polycurves. Figure 1 shows the basic idea of the lamp model. Notice the major parts that make up the lamp—base, supports, glass shades, and so on—and think about the types of materials these could be built from.

The Work Environment For the rest of the tutorial, we'll use the form·Z Modeling module. Switch back to the Model window by selecting it in the Windows pulldown menu. Open the Window Setup dialog from the Windows menu and set the X, Y, and Z Grid Modules to 6 inches with six divisions. Open the Snap Options dialog (also in the Windows menu) and set the X, Y, and Z Grid Snap Modules to 1 inch. This will define a 6-inch grid subdivided into 1-inch increments, which will be convenient for the size of objects in this project. Use the Zoom tool in the Window Tools palette to zoom in so that a grid area of about 6 feet by 6 feet fills the window. Select Graphic/Keyed from the Heights menu. This will allow you to graphically extrude objects, providing a more fluid, interactive method of creation.

Since the lamp has a number of components made of different materials, you need to set up a preliminary set of surface styles to distinguish them. For now, only change the names of existing styles to correspond to different parts of the model. We'll change the material properties later. Double click on the Style 1 icon in the Surface Styles palette to open its Surface Styles Parameters dialog. Change the style's name to "base." Change Style 2 through

Style 5 to "switch," "support," "glass," and "hardware," respectively.

Click on Layer 1 in the Layers palette and change its name to "lamp." For now, we'll place the objects on this layer and add additional layers as necessary.

Making The Base To create the model, we'll begin with the base and work our way up. I will include the approximate dimensions as I go, but feel free to alter them as you develop your own design.

Because the lamp is symmetrical about its vertical center axis and several parts will be made by copy-rotating them about this axis, we will center the lamp at the origin (0,0,0) to make construction easier. The center axis will be the Z axis.

Make sure the "base" surface style is selected. Select the Polygon tool and use the Tool Options palette (or the Polygon Options dialog in v. 2.9) to set the following parameters. Change the number of segments to 12, click on Polygon Pattern, and select the top left pattern. These settings will enable you to draw a 12-sided, star-shaped base.

Make sure the 3D Extrusion modifier is selected and make the base with its center at the origin. I used an outer radius of 1 foot, 6 inches, an inner radius of 6 inches, and an extrusion height of 4 inches. Locating the radius points along the Y axis will orient the star with the X and Y axes, making subsequent modification easier. Turning on Grid Snap will help in locating the points. Finally, taper the base using the Draft Angle tool (the default settings are fine). The blue object in Figure 2 is the tapered base.

Adding The Switches Next, you need to add six switches on top of the base. Select the "switch" surface style. Use the Move Plane tool (relocated to the Window Tools palette in v. 3.0) to move the reference plane grid up 4 inches so that it's aligned with the top of the base. Turn on the Perpendicular Switch to ensure the grid moves straight up along the Z axis.
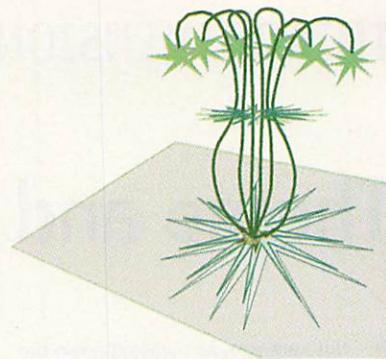
FIGURE 1. A quick study sketch of the lamp made with the form·Z Drafting module.

A

B

C

D
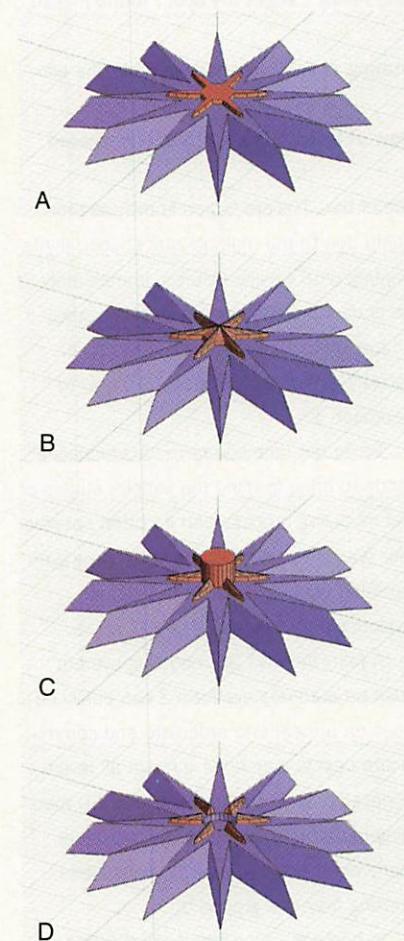
FIGURE 2. Four steps in the creation of the base and switches of the lamp. The Boolean Union and Difference tools are used to weld together and carve out parts.

To make the six-sided extruded star shape, select the bottom right Polygon Pattern to obtain a rounded edge, and use Construct Through Segment to align it with the base. Taper the shape (Figure 2A). This time you will need to adjust the settings in the

1.800.351.7620 www.caligari.com/ads/3dd

See the difference for yourself

trueSpace™4
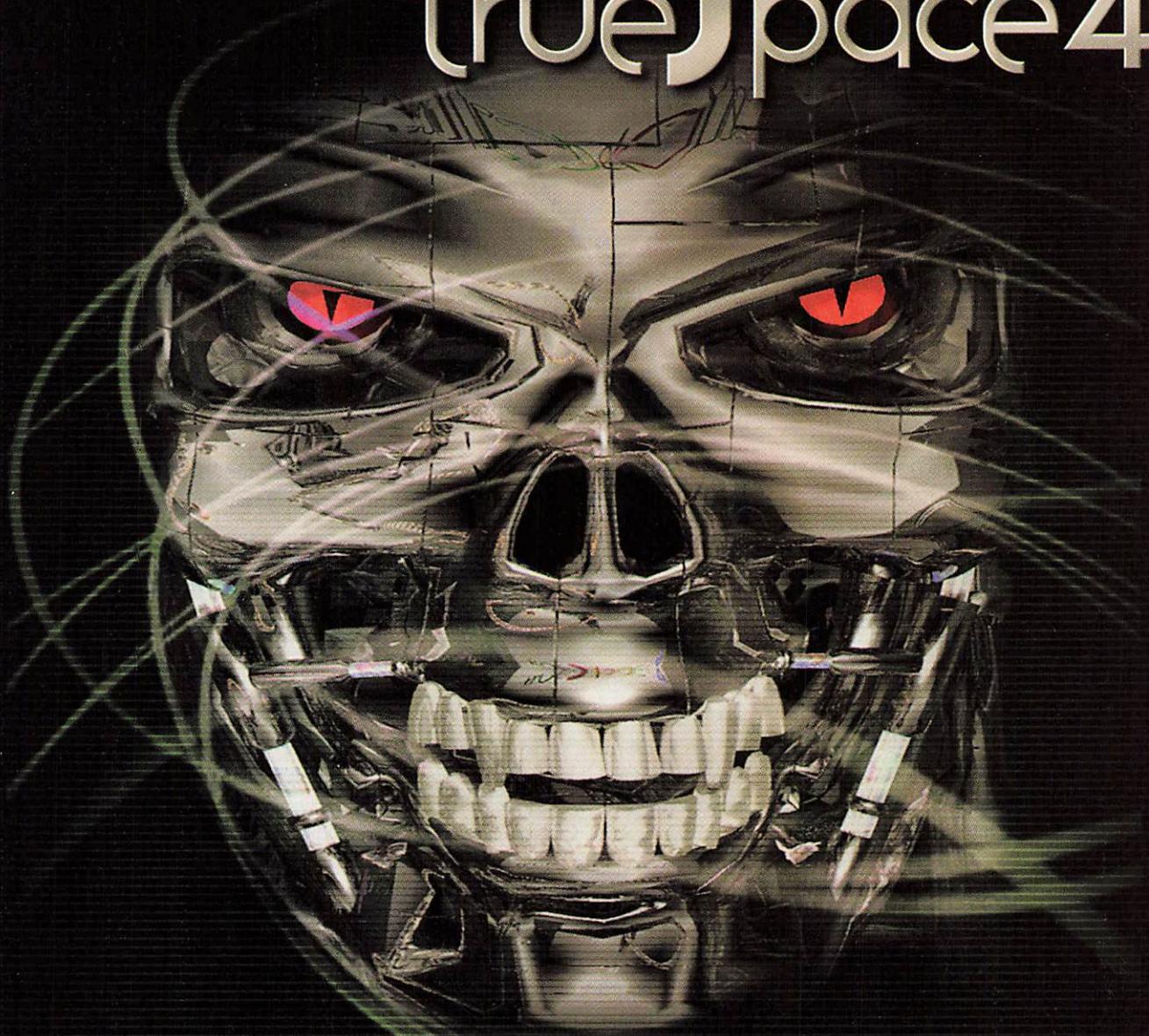
True 3D interface • Game quality acceleration • Inverse kinematics • Forward

dynamics • Prototyping with scripts • Precise collision • NURBS • Bones •

Atmospheric rendering • Reflectance shaders • Hybrid radiosity

$595

30-DAY MONEY BACK GUARANTEE

THIRDDIMENSION

Tool Options palette (or Draft Angle options dialog). Set the Position Of Base Plane to At Minimum Of Object/Faces (this will ensure that the taper begins at the base of the object, keeping the bottom face constant). You may also need to set a different taper angle since the object is relatively thin. I used 15° for my 1/2-inch extrusion height.

Next, select the Face Topological level. The Topological level modifiers determine the type of entity that subsequent operators will apply to, which allows for multiple levels of entity control. For example, the Draft Angle just done was applied to the entire object (the default Topological level is Object); the next operation will extrude just the top face of the switches. Select the Derivative Convergence tool and check the Keep option (Tool Options palette in v. 3.0 or Status Of Objects palette in v. 2.9). This will ensure that the original object is not deleted after the face is extruded. Select the top face of the switch by picking two of its edges and extrude it up about one inch. Now select the Union tool and pick the two parts of the switch to make them a single object (Figure 2B).

Carve out the middle part of the object to make six separate switches. Make an extruded circle at the center of the base using the 3D Extrusion modifier and the Circle By Center & Radius tool (Figure 2C). The reference grid should still be at the same level as the top of the base. The radius has to be at least as large as the inner radius used to make the original switch. The exact height is not important; it just needs to be taller than the top of the switch. Using the Difference tool, subtract this cylinder from the switch to make the six separate switches. Similarly, cut a depression into the center of the base for the supports. I used a height of -1 inch for the hole (Figure 2D).

Some design alternatives include trying variations of the base and switch shapes, for example, different polygon patterns and tapering angles.

Making The Supports The supports consist of six curving tubes (the rays) projecting from the hole in the base. Select the "sup-

port" surface style. Select Front view from the View menu and zoom/pan until about six feet of space is available above the base. The support is made by drawing a path curve, making a cross section, then sweeping the cross section along the path. I made the path a 2D Surface Vector Line and smoothed it with the C-Curve tool using the following options: a Deep Tangent curve and a 2-inch Smooth Interval (Figure 3A).

Making the initial shape as a 2D Vector Line rather than a 2D Spline (or Smooth Line in v. 2.9) reduces the number of points in the line. The main advantage of the tangent curves is that the curve passes through the original control points. Once constructed, the curve can be edited by selecting it again with the C-Curve tool. You should also try the Quick Cubic and Quick Quadratic curves for comparison (they will be smoother but may require more adjustment). You may need to rotate the curve 30° around the center axis to line it up with one of the switches. A Top view will clearly show the orientation.

To make the 2D cross-section shape, a small circle, hexagon, or star are all suit-

able. I used a five-pointed star; this shape will influence the form of the lamp socket that will attach to the top end of the ray. Use the Sweep tool to extrude the cross section along the path (Figure 3B). Use the Rotate tool with the Multi-Copy modifier (number of copies set to five) to copy the support around the lamp using a 60° rotation about the origin (Figure 4).

You might want to try different path curves and cross section shapes. The six rays do not need to be identical; try alternating patterns made of two or three shapes.

Examining Materials Before creating the lampshades, let's experiment with a few materials. We will define material properties for the "base," "switch," and "support" surface styles. Double click on the icon in the Surface Styles palette to open the Surface Style Parameters dialog. I often begin by selecting one of the predefined materials as a starting point and then modify the parameters. Click on the Predefined button to open the Predefined Material dialog, choose a category, and select a material by clicking on its icon. I used Cast Aluminum

A

B

FIGURE 3A. A front view showing the control points of the path curve and the star shape that will be swept along it to form a support. The curve is tweaked into shaped by dragging the control points. The vertical line indicates the axis of rotation (Z axis) about which the support will be copy-rotated.

FIGURE 3B. The Sweep Preview dialog showing the sweep parameters and resulting object.

FIGURE 4. (left) The six supports formed by copy-rotating one around a vertical axis through the center of the base.

THIRDDIMENSION

for the base, Bronze for the supports, and Rubber for the switches. Then, using the Options buttons in the Surface Style parameters dialog, I reduced the amplitude of the aluminum bumps to make a smoother surface and changed the color of the rubber to a yellow.

To add a floor pattern, define a new surface style named "floor," set its color to Image Map, and load one of the predefined floor patterns from the textures folder in the form·Z folder. I used "floor1," but there are a number of variations of this tile pattern. Select the "floor" surface style and make a very thin horizontal shape beneath the lamp (move the reference plane grid back to Z = 0 and extrude down). Select RenderZone from the Display menu to produce a rendering. Figure 5 shows the lamp including the glass shades we will model in the next section.

For some design alternatives, you can try different material properties for the various parts of the lamp. Because this process often requires making many small changes and re-rendering, it's useful to check Set Image Size in the RenderZone Options dialog. This will allow you to graphically pick a small part of the image to render, making the process much faster than rendering the entire model and allowing side by side comparisons between different renderings. Using texture image maps instead of predefined materials will often produce more detailed results; I tried "galvanized" for the base and "board1" for the floor (both found in the textures folder).

The Glass Lampshades Starlight lampshades typically come in two flavors: stellated dodecahedrons/icosahedrons and "inverted star-bowls." The former shapes were derived from Platonic solids in the early 17th century by astronomer Johannes Kepler and have been the design inspiration in recent times for lampshades and ornaments. Although these shapes are mathematically complex, they are quite simple to model. The inverted star-bowls, reminiscent of the undulating glassworks of the Finnish architect Alvar Aalto, are slightly more com-
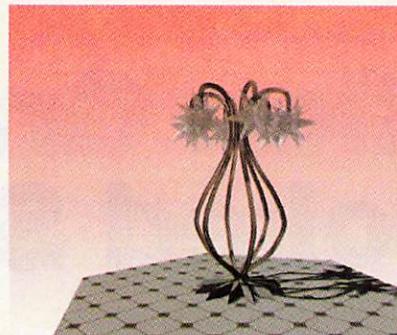
FIGURE 5. A rendered image showing preliminary material properties of the various parts of the lamp. RenderZone options were set to Full Z-Buffer with Shadows and Graduated background.

plex (but more fun) to model. We will model some of each.

First the Platonics: Create a new layer named "shade1" and make it current. Select the "glass" surface style and make a dodecahedron and an icosahedron using appropriate options of the Spheroid tool (Spherical Solid tool in v. 2.9). Construct them off to the side of the lamp using a radius of about 4 inches. We will move them into place later.

Typically, a 2D surface or one face of an object is extruded into a 3D form (as you did with the top of the switch), but you can also extrude all the faces of an object at once to produce a more complex object. Select the Derivative Convergence tool and set the Perpendicular To Surface option so that each surface extrudes perpendicular to itself rather than to the grid plane. Also, select Single Object (Tool Options or Status Of Objects) so that the result of the extrusions is one object rather than many separate objects. Finally, select Custom from the Heights menu and set the height to 8 inches so that all surfaces extrude to the same height at once. Pick the two objects to extrude them (Figure 6).

Now, copy and move the shades into place. We will alternate three of each shape

sources
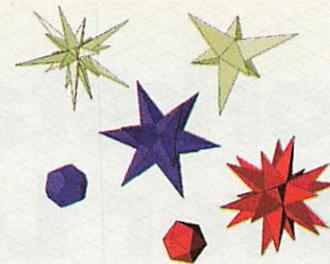
FIGURE 6. A (blue) dodecahedron and (red) icosahedron (12-sided and 20-sided Platonic solids, respectively) and the extrusions into their stellated forms. In the background are examples of more complex lampshades made by varying extrusion heights of individual faces.

around the lamp. In a Top view, move each shade until it's centered on the end of a support (use adjacent supports), then in a Front view, move them up into place. Select the Pick tool and prepick both shades. Using Multi-Copy (set to 2 copies) and Rotate, copy the two shades 120° around the center axis to the other supports.

Define the "glass" surface style: Select the predefined Glass material and change its transparency to Simple or Center-Edge. Try a rendering. I used RenderZone with a graduated background and transparent shadows—two options that will significantly increase rendering time. If rendering is very slow, try changing the "glass" surface style's reflection from Glass to Constant or Chrome.

To create the second type of lampshade, make a new layer named "shade2" and turn off layer "shade1." To achieve this shape, we will create a framework of boundary lines and wrap a skin around them. Select the 2D Surface modifier and use the Polygon tool to make two concentric stars—one with an inner radius of 6 inches and outer radius of 10 inches and the other with radii of 1 inch and 2 inches, respectively. Move the smaller star straight up 10 inches. Use the C-Curve tool to round the points of the stars. I used Shallow Tangent with smooth intervals of 1 inch and 1/4-inch for the large and small stars, respectively. Draw a 2D Vector Line between corresponding points of the two stars (use point snap to ensure the line is exactly on). You can use a Quadratic Bezier Spline (or Smooth Line in v. 2.9) instead to make a

more flowing profile. Copy-rotate the line about the center of the star to the other points (Figure 7A).

Now you're ready to wrap the skin. Set the Skin tool to skinning along paths with two sources (the stars) and five paths (the profiles). Be sure to select the sources first then the paths (Figure 7B).

The top of the shade is made by skinning between the top star (un-ghost the original) and a circle above it (Figure 7C). Stitch both objects together with the Trim/Split tool (this tool is similar to the Booleans, except it works on surface rather than solid objects). Move and copy the shades into place as before.

After considering the two lampshade alternatives, I decided to continue with the second. The next step is to make a fixture to hold the glass shade in place (Figure 7D). This is done by extruding the end face of the support, then scaling its lower face enough to enclose the top of the shade. Next, use the Round tool to fillet the corners (Figure 8). Finally, use the Difference tool to cut out a cylindrical opening through the fixture and place a socket in it using a 3D Enclosure.

You can try some variations on the lampshades by using the Graphical/Keyed Heights option to graphically extrude each surface of a Platonic form independently, resulting in a shade containing some long and short rays. Modify source and profile shapes to produce variations of the skinned shade. Using different layers will enable you to quickly switch back and forth between the alternatives by turning the layers on and off.

Setting Light Sources Now its time to experiment with lights. First, we'll add light sources inside each of the six glass shades. Click in the Lights palette to create a new light, change its name to "bulb1" and double click on its name to open the Light Parameters dialog. Set its Type to Point light (to simulate a single bulb) and its Falloff to Square. Assign it a color of your choice and set its Intensity to about

A

B

C

D
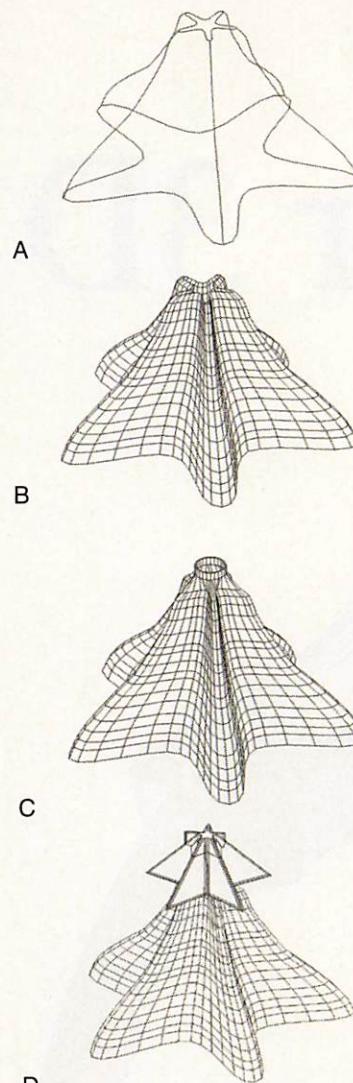
FIGURE 7. The steps in making the inverted star-bowl shade. A. The source shapes and profile lines are drawn. B. The skin is applied. C. The top part is similarly made and stitched to the first part. D. Finally, the fixture to hold the shade is made.

50 percent for starters. Finally, change its radius to about 6 inches, or small enough to be contained within the glass shade. Exit the dialog, and use the Move tool to position the light into place inside one of the shades. As with the shade, this can be easily accomplished using a Top and then a Front view. Now copy-rotate it so that each shade contains a light.

At this point, you may want to edit the names of the lights and give them different colors to enhance the lighting effect. Set an appropriate view and use the Full Raytrace option of RenderZone (Figure 9). The addi-
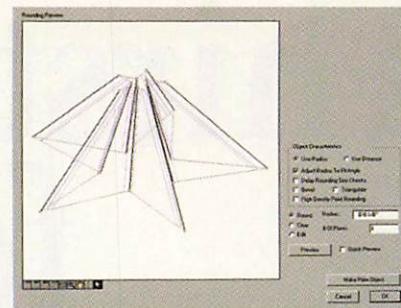
FIGURE 8. The Rounding Preview dialog showing the rounding parameters and a preview of the object (a part of the fixture). The ability in v. 3.0 to edit the rounded object with the same dialog is a great help for design exploration.

FIGURE 9. A view of the lampshades with light sources in place. The glass shade material is made by changing the predefined glass surface style to chrome reflection, center-edge transparency, and lightening the color.

tional light sources combined with transparencies will cause a significant increase in rendering time. To run quicker test renders, you can turn some of the lights off by clicking on the Shining icon (black dot) in the Lights palette. You can also turn off the default Sun light if it adds too much light to the scene and add cone lights to highlight particular areas or cast shadows.

It requires a lot of experimentation to achieve your desired lighting effects. Try adjusting the light parameters and do several more renderings. Point lights emit light rays of equal intensity in all directions. By using a cone light, you can direct the light

rays in a particular direction, such as toward the base of the lamp.

Although RenderZone produces reasonable-quality renderings (certainly good enough for design development), you'll need to use form·Z Radiosity or export your model to another rendering application to get really slick renderings.

Details, Details Now that we have a reasonable working model, it's time to bring into play the spirit of the architect Mies van der Rohe (who is attributed with saying "God is in the details"), and work on the lamp's details. Continue the development of the model according to your own design ideas.

A few items to consider are the decorative support connecting plate, rubber gaskets around the switches, power cord, lightbulb, and additional hardware.

By now we have a good sense of using a modeling system as a means to engage our design ideas. It's easy to get caught up in the desire to make beautifully rendered finished models, and there is nothing wrong with this except when it comes at the expense of exploring ideas and experimenting with alternative possibilities. It is sometimes useful to remember that a modeling system is a fluid, dynamic medium that can and should be used a little more like clay and a little less like bits and bytes. ⬤

Lucien Swerdloff consults, teaches, writes, draws, and designs with clay and bits. His book, *form·Z by design: a brief tutorial* (Stipes Publishing, Champaign, IL, 1997), presents a design-based introduction to modeling with form·Z. He lives somewhere between the mountains and the sea in the Pacific Northwest.

what's new in form·Z 3.0

form·Z 3.0 is a major upgrade containing some significant new features. Key enhancements include a new, more flexible interface and new and improved modeling, drafting, rendering, animation, and import/export tools. Following are descriptions of some of the primary enhancements:

Redesigned Interface The interface, as usual with form·Z, is well designed and sensibly structured; however, as with many other programs, the screen area devoted to tools and palettes keeps getting bigger as more functionality is added. Interface enhancements include:

· The default toolbars are arranged into two columns instead of one. Toolbars can now be customized by selecting Customize Tools in the Palettes menu.

· A new context-sensitive Tool Options palette allows settings for the current tool to be changed directly without invoking a dialog (although the dialog can still be opened, as in previous versions, by double clicking on the tool). The Status Of Objects palette no longer exists because its functions have been incorporated into the Tool Options palette.

· Preview dialogs for many tools, including Sweep, Revolve, Screw/Bolt, and Round, have been redesigned to allow easy inspection of results and subsequent editing of parameters.

· A new Animation palette has been added to update and play animation.

· The ability to draw across windows and snap to different types of entities simultaneously has been added, improving object construction.

· All of the reference plane (grid) tools have been moved from the toolbars (where they always seemed out of place) to the Window Tools palette (where they should be). The Drafting Module now supports moving and rotating of the reference plane. (Along with the new layout capabilities, this will greatly enhance the ability to compose plots.)

Modeling Enhancements Although there are far too many modeling enhancements and new features to discuss here, some of the highlights include new
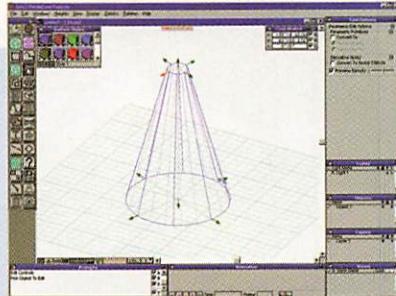
The form·Z v. 3 default interface. The graphics window shows a parametric object with controls that can be used to graphically transform the object.

spline, NURBS, and patch modeling techniques, many editing improvements and improved skinning and stairs tools, as well as drafting enhancements.

The big news is the addition of parametric objects. In keeping with recent trends in 3D modeling, form·Z now supports the generation of parametric primitives (such as block, cone, sphere, and torus). A new set of editing tools called Edit Controls and Edit Surface allows easy manipulation of object parameters. For example, cones can be truncated, blocks can be extended, and geodesic spheres can be converted into revolved spheres with a few clicks.

In addition, derivative objects, such as sweeps and revolutions, are now parametric so they can be more easily changed than in previous versions. These and other parametric objects can still be combined with other types of objects (although they may lose their parametric status).

Although you will not find advanced parametric and feature-based objects (such as constraints), the addition of parametric objects to the already robust set of object types that can be modeled ensures that form·Z will remain at the forefront of 3D modeling systems and be in a position to add full parametric modeling in future versions.

Animation The ability to do animation has long been a desire of otherwise content form·Z users. Version 3.0 has the ability to produce simple fly-by animations, render and replay sequences, and export to AVI and QuickTime formats.

—*Lucien Swerdloff*

© Copyright Fox Sports. Image courtesy of Digital Dimension.

"We needed to find a way of dealing with the huge computing loads that serious ray tracing presents..."

The Super Bowl is the single biggest event in broadcast television, with up to 800 million viewers worldwide. This year, Fox Sports was looking for someone to produce a visually stunning animated opening sequence for "*Super Bowl XXXIII on Fox*". They turned once again to Emmy award-winning Digital Dimension, a North Hollywood-based studio specializing in 3D animations, motion graphics, and visual fx.

Digital Dimension faced a number of challenges. First of all, the project needed to be completed rapidly. With a short time to complete the job, and lots of creative interaction with Fox expected, speed was of the essence. Secondly, while much of the sequence could be done using conventional tools and methods, the

inclusion of large stretches of water and a gigantic replica of the Vince Lombardi Trophy meant that they would have to be ray traced in order to achieve the level of photorealism required.

Normally, installing the rendering resources on the scale required by a project of this complexity would have been a difficult, expensive, and time consuming process. But thanks to RenderDrive's revolutionary rendering technology, it proved quick and trouble free.

Fox Sports loved the finished results, and Digital Dimension's animators loved the quality and power of the RenderDrive. In fact, they may need to break out their dress shoes again for this year's Emmy awards ceremony.

"We could not have done this project in the time allotted and to the quality standard that Fox requires without the power, speed and increased interactivity that RenderDrive gave us."

Ben Girard
General Manager, Digital Dimension

Ray tracing finally makes sense

RenderDrive

Dem Bones

How to build lifelike animated characters using Hash Animation:Master 99.
by Raf Anzovin

Advanced tools for boning and moving animated characters have developed rapidly in high-end animation software over the past couple of years, but they've been slow to appear in lower cost packages—with the notable exception of Hash Animation:Master 99. With nearly a decade of development behind it, this $199 program, available for Macintosh and Windows NT/95/98, has matured into a powerful environment for character articulation and animation, with features that rival those in higher priced apps such as Softimage and Alias|Wavefront Maya.

Animation:Master 99's boning system allows fine control and pinpoint accuracy in mesh deformation. The new Smart Skin system makes it easy to control complex bulging, stretching, and wiggling. Moreover, bones can be constrained in complex ways by the position or orientation of other bones. Setups can be made to perform smart functions, such as automatic center-of-gravity placement, that require complex expressions in other programs.

This articulation system makes Animation:Master 99 suitable for constructing all kinds of characters, both cartoon and realistic. In this article (and its follow-up companion available on the *3D Design* web site at www.3d-design.com), we'll work step-by-step through the setup of a cartoon character, Dennis the Dog (star of *Java Noir*, winner of the "Best 3D Cartoon" category in *3D Design*'s 1998 Big Kahuna Design & Animation Contest). This new version of Dennis (whose *Java Noir* incarnation was created in NewTek LightWave) has been modeled with an unusual goal in mind: to combine cartoon proportions with semi-realistic anatomical structures. Dennis perfectly demonstrates the complex interplay between skin, muscles, and bones that is one of Animation:Master 99's strengths.

This article covers basic bone and constraint techniques, including joint construction, Smart Skin, and facial boning. In the web supplement, you'll find more advanced aspects of boning, including control bone systems, automatic center-of-gravity placement, and a way to create a single bone structure that can quickly switch between low-res and high-res skins. Both installments assume familiarity with basic Animation:
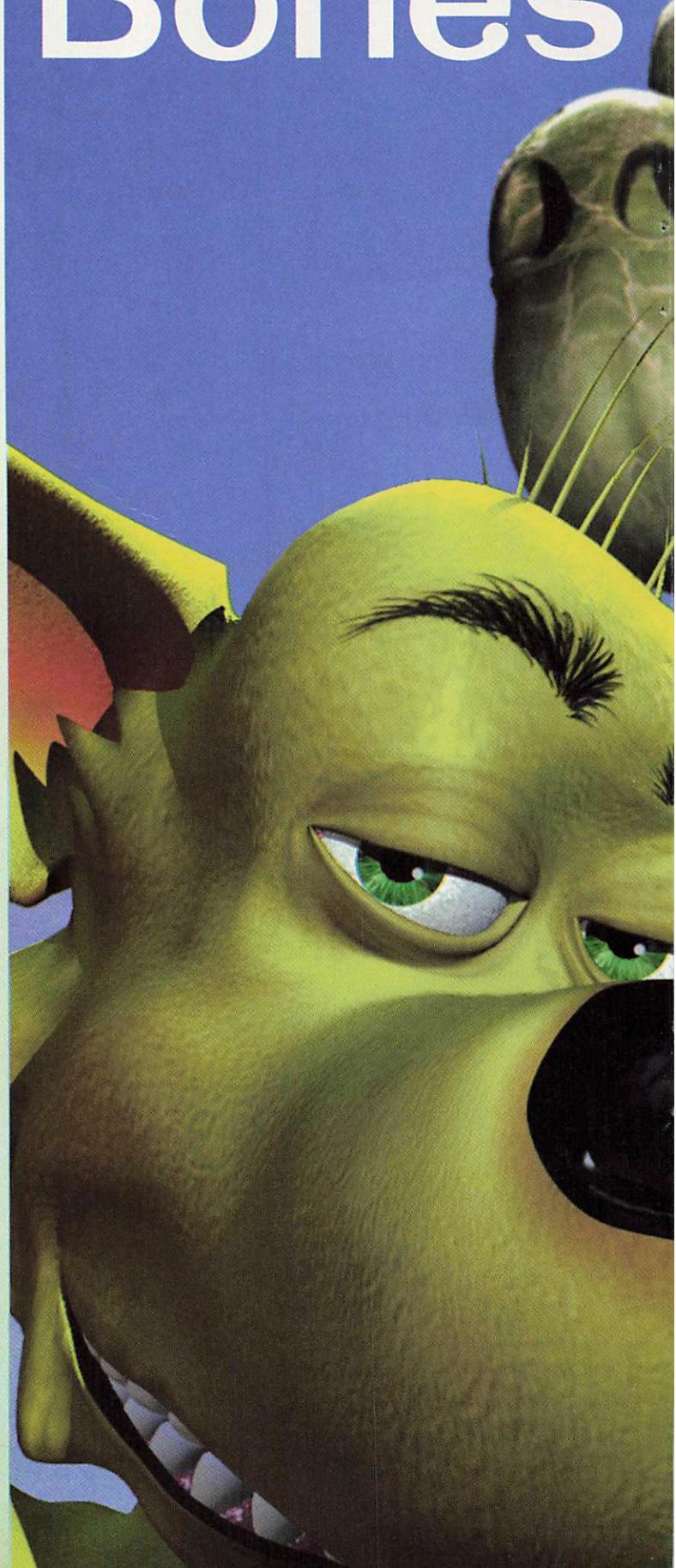
Dem Bones

Master 99 concepts such as Hash splines, control points, patches, constraints, Pose Sliders, Actions, and Choreographies, as well as how to create a bone and attach it to a spline.

Dem Bones is Gonna Rise Some types of character design are more amenable to bone manipulation than others, so it's important to think about how you're going to bone a character before you model it. This is most important in the joint areas, where most of the mesh deformations happen.

Examine the spline mesh that makes up Dennis's surface (Figure 1). Dennis is designed to facilitate accurate boning and twisting of the mesh. Areas that are meant to be flexible and bendable are distinguished from areas that are meant to be rigid by the spline structure that forms the surface of the mesh. In most areas, care has been taken to shape the surface as if real bones, muscles, and ligaments were under the skin. His head, for instance, is built around a cartoon skull structure that combines features of human and canine skulls.

In the deltoid muscle area (the top of the shoulder), notice that the splines are bisected three times by cross sections. These cross sections are added to facilitate the intermediate bone technique that we'll look at in a moment. The same spline structure is used in the finger joints and many other places.

Boning in Animation:Master 99 is simple and intuitive. You attach individual control points (CPs) to each bone by hand, much the same way clusters are used in high-end programs. (Unlike control vertices, or CVs, of a conventional NURBS surface, CPs of a Hash spline are attached directly to the curve; thus the distinction.) CPs are parented right to the bone; joint flexing and bending does not occur automatically. For some very simple joints—Dennis's neck, for instance—this is all that's needed, and the splines can be trusted to create a smooth bend. Most joints, however, require complex bending and bulging, which means more complexity in joint construction.

We'll use two different methods to construct smooth, working joints. One is the intermediate bone method, which involves adding an extra bone between two bones in a joint—for example, an extra wrist bone between the hand and lower arm bones. The CPs in the middle of the wrist are attached to this intermediate bone. The intermediate bone is then given two Orient Like constraints, one to the hand and one to the lower arm. Hash's Orient Like constraint takes the absolute rotation value of one bone and transfers it to another bone, pointing both of them in the same direction irrespective of the parenting structure. Animation:Master 99 automatically averages the rotations of both bones and applies the average to the wrist intermediate bone, creating a joint that bends smoothly, without crimping. This system is easy to set up and works well with bends in any direction; however, the joint tends to look and move like rubber tubing.

The second jointing method takes advantage of Animation:Master 99's new Smart Skin feature. Smart Skin provides excellent control over the way joints bend. You can animate and morph CPs freely, then link your animation to the rotation of the bone so the animation plays back as the bone rotates (similar to the Set Driven Key functions in Maya). Setting up a Smart Skin joint is more time-consuming than cre-
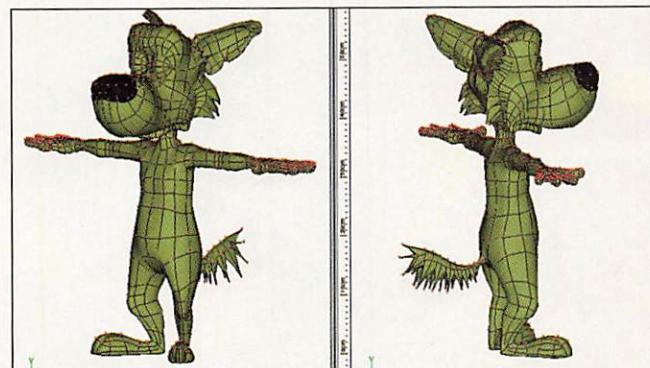

FIGURE 1. The splines of Dennis the Dog's 2,000-patch mesh are carefully placed to facilitate boning and twisting.

ating a simple intermediate bone joint—you have to specify deformations for the X, Y, and Z axes independently if you want a joint to look good as it rotates in any direction—but the results are well worth the effort. You can also use the techniques together and layer more than one deformation, as we'll see when we assemble the shoulder joint.

Arm Bone Connected to the Wrist Bone Let's start with the basic bones of the arm. Using the Add Bone tool, click in the shoulder area and lay down a bone that stretches from somewhere near the neck to the middle of the deltoid muscle. Click again on the end point of the pectoral bone and create another bone that ends in the elbow. Make sure that the Attached To Parent Bone option is checked in the Properties panel to ensure that the two bones are linked together so pulling on the joint between them affects both bones.

Repeat this process to add two more bones, one ending in the middle of the wrist and the other extending outward some distance past the end of the hand. The extra length makes the bone easy to grasp when you're animating the character in a zoomed-out view. Figure 2 shows the basic arm bones.

While the wrist is actually one of the more complex joints in a real body, the surface deformation it represents is basically a simple bend. For that reason, you can use the intermediate bone technique.

Zoom in on the wrist area and select the lower arm bone. Add an intermediate bone, a child of the forearm bone. Place the beginning point of the intermediate bone directly on the hand/forearm joint and make sure it's pointing in the same direction as the hand. It's important to make sure the roll pointers are pointing in the same direction.
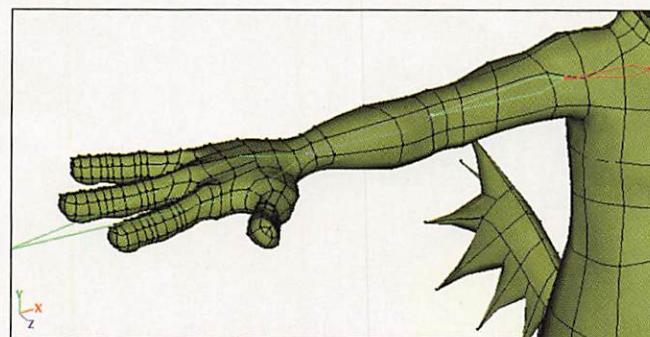

FIGURE 2. Dennis has four basic arm bones: hand, forearm, lower arm, and an extra shoulder bone used for shrugging.

You'll want to average all rotation, including twisting the bone along its roll axis. Failure to put the roll handles in the right direction will mangle the mesh when you apply constraints.

This procedure will create a good bend on all axes, but it's only the beginning. The roll axis—the axis that represents the wrist twisting rather than bending—still needs quite a bit of work. In a real wrist, twisting occurs in the forearm, not the wrist itself. To create this effect, you can add more bones that will be used with Roll Like constraints (which are similar to Orient Like but transfer only the bone's roll axis value, not the entire rotation) to spread the twisting deformation down the length of the forearm while leaving the bending axes untouched. Actually, we're going to cheat a little by allowing the wrist joint to take some of the twisting. Odds are that no one will notice the slight anatomical liberty, and it makes the twisting less noticeable in the arm.

Add two more bones parented to the lower arm, placed directly over the two cross sections near the wrist. Call the bone nearest the wrist Twisting Bone 1 and the other Twisting Bone 2.

Using the Group tool, assign CPs to all the bones. The lower arm bone gets the CPs right before the elbow, the hand gets the all the CPs in the hand up to the wrist, the intermediate wrist bone gets the cross section lying exactly on the wrist joint, and the two twisting bones get the cross directly beneath them. Figure 3 shows the completed wrist bone structure.

Now, create a new pose for the model. This pose will hold the con-
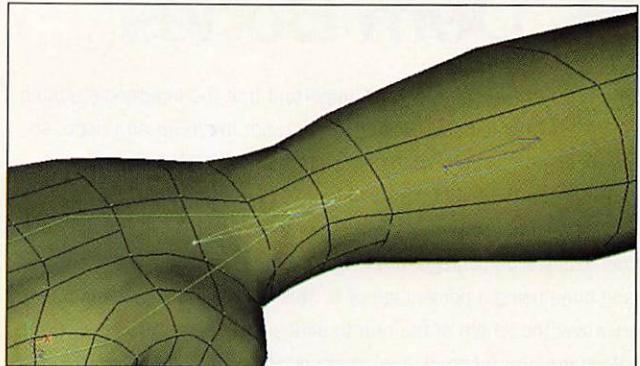
FIGURE 3. The wrist intermediate bones. You can see which CPs have been attached to which bones by their color.

straint information, which you'll apply to an Action later. At that point, you can use Animation:Master 99's Pose Sliders to test the setup. Pose Sliders are similar to the Custom Attribute feature in Maya. They allow any number of animatable attributes—bones, CPs, constraint influence values—to be driven by the movement of a single slider in the Pose Slider window.

Select the intermediate wrist bone and apply an Orient Like constraint; constrain it to orient like the hand bone. If you've oriented the intermediate bone correctly, you should see very little change when it snaps into a new position that reflects the new constraint. Now apply a second Orient Like constraint, orienting the bone like Twisting Bone 1. Applying two Orient Like constraints in a row tells Animation:Master
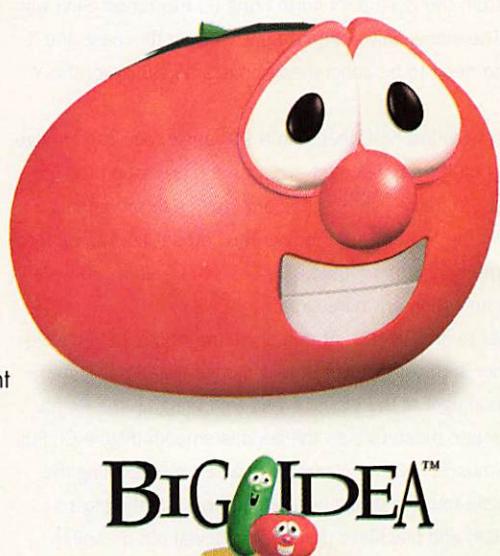
Dem Bones

99 to average them. It's important that the intermediate bone is oriented like Twisting Bone 1, not the lower arm bone, so the twisting of the arm will be consistent and smooth.

To spread the roll of the hand bone to Twisting Bones 1 and 2, use Roll Like constraints. Set Twisting Bone 1 to roll like the hand bone, using a percentage of 45. Set Twisting Bone 2 to roll like the hand bone using a percentage of 9. This will make the twisting effect taper over the length of the arm to almost nothing at Twisting Bone 2, making the arm respond in an anatomically plausible way when the wrist is twisted.

To test this setup, make a new Action and bring up the Pose Slider window. A slider will appear for the pose you just created. Set the slider to 100 percent to apply the constraints. (At this point, no constraints will appear in the Project Work Space, which is a File Manager-type window that lists objects, groups, constraints, and so on.) Use the Lock tool on the hand bone to keep it from trying to calculate inverse kinematics (IK), then rotate it all around. Make sure to rotate it along the roll axis to check the twisting effect (Figure 4).

Also try moving the bone around a little in the modeling window. Change the intermediate bone's center of rotation to adjust how smoothly or sharply the joint bends.

Elbow Bone Connected to the Smart Skin

Unlike the wrist, the elbow is an extremely complex joint. Dennis's semi-realistic design requires a clear differentiation between soft muscle areas and rigid bone. Moreover, when the elbow is sharply bent, the inner muscle surfaces must collide and bulge outward. In other words, the elbow is a perfect candidate for Animation:Master 99's new Smart Skin tools.

To begin using Smart Skin, right-click (or command-click on a Mac) the lower arm bone's name in the Project Work Space. Select X-Direction Smart Skin from the pop-up menu to bring up the Smart Skin window (Figure 5). (The elbow only bends on one axis, in this case the X axis, so there's no need to be concerned about Smart Skin for the Y and Z axes.)

The Smart Skin window first opens on a 90° angle. You can change the angle by dragging the frame count slider on the animation toolbar below. Notice that the frame you drag to corresponds to the bone's angle of rotation in the Smart Skin window. You can set keyframes for the CPs at any time during this rotation, just as you would if you were animating CPs in an Action or Choreography.

Move the slider to roughly 58°. At this point, the arm is bent considerably, but the inner surfaces don't yet meet. Select the CPs that make up the cross section right at the elbow. Press R to bring up the Rotate manipulator and rotate these CPs so the bend is smooth (Figure 6). Pull in the CPs of the muscle areas and pull out the CPs representing the elbow tip to simulate muscle sliding over bone. Then pull the biceps (upper arm muscles) and brachialis (forearm muscles) out as well to simulate the effect of muscles bulging and tightening to pull the arm. You can test the effect by dragging the animation slider back and forth.

Continue sculpting the arm in this way as it moves. An arm should be able to bend at least 150°, as in Figure 7. While you may never bend it that far when animating, it's often necessary to push a model
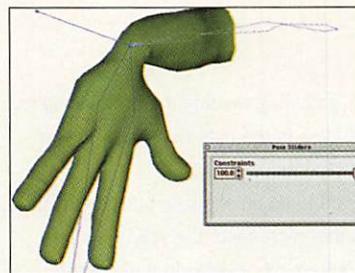

FIGURE 4. The wrist tested in an Action. Notice that the wrist is both bent and twisted, and that the twist spreads down the arm.
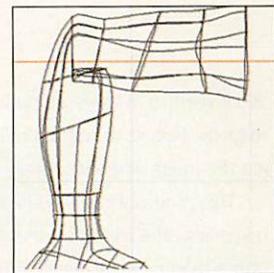

FIGURE 5. The Smart Skin window showing the elbow joint before applying Smart Skin deformations.

farther than real human anatomy allows. This means quite a bit of muscle collision and bulging as the two arm surfaces press against each other. To keep the muscle volume constant, you'll have to bulge them out a lot.

To test the Smart Skin effects, create a new Action, lock the lower arm bone to disable IK, then pull the bone around. The Smart Skin animation takes effect immediately. (Note that the arm is not yet meant to be rotated in any axis but the X axis. In the web-based portion of this tutorial, you can read about how to set up the arm for full mobility in a way that will ensure that the elbow always points in an anatomically correct direction based on the curve of the arm.)

Smart Skin & Intermediate Bones Combined

Smart Skin and intermediate bones can be used together, as we'll demonstrate with the shoulder joint. The deltoid area needs to be able to rotate in any direction and bend smoothly, making it a good candidate for the easy-to-set-up intermediate bone technique. At the same time, the underside of the upper arm must undergo complex morphing as it nears the side of the body, a job for Smart Skin. Combining the two allows us to set up the joint more quickly than using Smart Skin alone.

The shoulder joint is constructed much like the wrist joint. The only difference is that the roll-twist is tapered down the arm from shoulder to elbow. Therefore, the two twisting bones must be parented to the shoulder bone, with their center points at the shoulder joint. Since they won't naturally rotate the same amount as the upper arm bone, both Orient
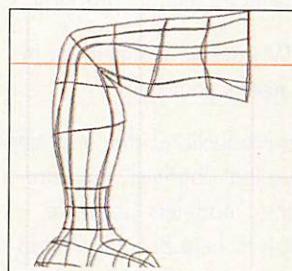

FIGURE 6. After using Smart Skin to sculpt the elbow deformations, a 90° bend not only looks smooth but can be made to suggest the interaction between bone and muscle.
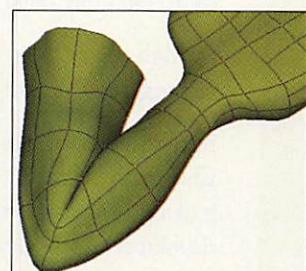

FIGURE 7. The arm at an extremely oblique angle. Many boning systems will break down at this point and make the mesh look like crimped tubing—but not with Smart Skin.

Rapid 3D #16

Dem Bones

Like and Roll Like constraints will be required to hold the arm together. Figure 8 shows this bone setup and its effect when the shoulder is bent.

To create the Smart Skin deformation, you need to see more than the Smart Skin window can show—you need to see the deformations created by the intermediate bone setup and twisting bones as we work. Therefore, you need to do all the deformation and point animation in an Action window and copy the keyframes manually into the Smart Skin window.

Open a new Action window. Drag the Constraints Pose Slider to 100 percent to apply all constraints. Lock the forearm and lower arm bones to disable IK. Now go into the Rotation tab in the upper arm bone's Properties panel and change Edit Angle As from Vector to Euler. This will change the rotation readout to X, Y, and Z values (that is, Euler values) that can be copied manually into the Smart Skin window.

The underside of the upper arm needs Smart Skin deformation where it comes into contact with the body as the arm is dropped down. Rotate the upper arm bone down about 80° on the Y axis. Using the Rotate manipulator rather then the bone's handle will make it easier to work only within the Y axis. Go into Muscle mode to tweak the positions of the CPs.

When you've gotten the arm to look good at this extreme, open a Y-axis Smart Skin window for the upper arm bone. Look at the Y axis readout on the Properties panel and copy that number to the frame counter in the Smart Skin window. Now go back to your Action and delete the bone's rotation. What remains are the point transformations you created, but now they're unwrapped from the effects of the bone. Simply select all the points you've edited, copy, go to the Smart Skin window, and paste (Figure 9). You can repeat this process for other extremes of upper arm movement as necessary.

Dem Mirrored Bones

Once you've created bones, constraints, and Smart Skin setups for one side of a character, you can use Animation:Master 99's new Paste Mirrored feature to mirror them on the other side. Using Paste Mirrored saves a lot of effort—if you set it up correctly.

The first step is to copy all the bones to the other side. Make a root bone at the 0,0 point and parent all the other bones to it. Control-drag the root bone onto the Object name in the Project Work Space (in this case, Dennis), which makes a copy of every bone. Scale the root bone –100 percent on the Z axis only. Now rename every bone to be mirrored so its name specifies right or left, as in Right Hand and Left Hand. Paste Mirrored will not work correctly if you don't.
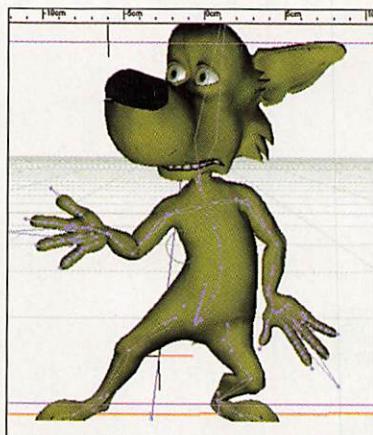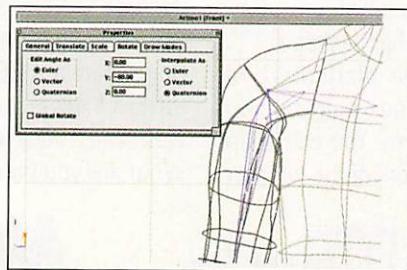
FIGURE 10. The fully boned Dennis. Paste Mirrored was used to transfer all the constraint and Smart Skin settings to the left side.

Once this is done, mirror the constraint set-up by going to the Constraints pose, copying, and choosing Paste Mirrored from the edit menu, making sure the Key Model button is depressed when you copy. To copy Smart Skin setups, open each Smart Skin individually and copy the point data out to a new Smart Skin for the corresponding mirrored bone, just as you did for the upper arm Smart Skin set-up. Figure 10 shows all the bones in the complete Dennis skeleton.

Boning for Facial Animation

The last topic in this installment is facial boning. In Animation:Master 99, a face is typically set up for animation using Pose Sliders, which you've already used to apply a set of constraints to a series of Actions while testing the arm's deformation. Pulling the Constraints slider to 100 percent fully applied the constraints.

When used to pose a face, Pose Sliders make it possible to generate a wide variety of expressions from a few controls. However, although we'll use Pose Sliders extensively in the web-based portion of this tutorial, we won't use them here. Dennis's facial setup is entirely dependent on bones, at least until we get to the final setup stage. The main reason is that, while Pose Sliders are great for driving animatable attributes, they can't be driven by anything themselves. There's no constraint that allows you to drive a Pose Slider value. When we do the control bone setup in the web supplement, it will be necessary to drive elements in the face by the rotation of other bones, even though a Pose Slider layer will be used for tweaking. Also, using bones for facial animation allows us to drop poses onto the model to change phonemes (the phonetic shapes that make up speech) and facial expressions, which isn't possible using Pose Sliders alone.

The secret to constructing an effective facial boning structure is to know that Smart Skin isn't limited to controlling the way the CPs act as a joint bends. It can link any point transformations to a bone's rotation. That means, for instance, that rotating a Smile bone could pull the CPs corresponding to the zygomatic muscles (along the cheekbone, from the

FIGURE 8. The shoulder bones in a resting position. The deformation has been created directly in Action by animating CPs, no Smart Skin yet.
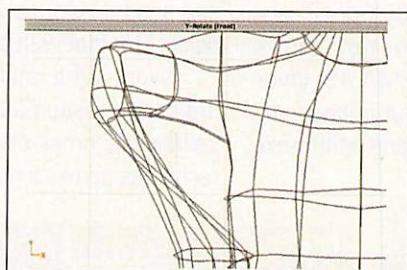
FIGURE 9. The arm looks distorted in the Smart Skin window because the constraints applied to the Action don't show up in Smart Skin.

corner of mouth to the temple) without actually attaching those CPs directly to the bone.

For other facial controls, such as the jaw, CPs are attached to a Jaw Rotation bone, but Smart Skin controls how the lips react to the lowering of the jaw. Dennis's head was actually designed to fit around an imaginary cartoon skull shape, so it's easy to find the pivot point of his jaw hinge and place the jaw bone there (Figure 11). You can then attach all the CPs from his lower jaw, including teeth and tongue.

Open a Smart Skin window for the jaw bone's X axis. At 90°, the face looks quite distorted. Dennis's mouth only rotates open about 15° before the ends of the jaw poke through the back of his neck (Figure 12). However, that doesn't mean you can't exaggerate the jaw opening beyond that point when the need arises. In this case, use

Smart Skin to translate the lower jaw points so the jaw can open wider. This distorts the jaw, but careful Smart Skin manipulation will keep it looking good. Once the jaw is open as far as it can go, manipulate the CPs in the lips to change the way they react to the opening jaw.

In the web-based installment (www. 3d-design.com), we'll set up this skeleton for animation using a few easy-to-pose control bones. We'll use Animation:Master 99 features such as Action Objects, Additive Actions, Pose Sliders, and a few undocumented constraint secrets to create a fast and efficient animation workflow. ●

Raf Anzovin is an independent 3D character animator. His animation *Java Noir* won the Best 3D Cartoon category in *3D Design*'s 1998 Big Kahuna Design & Animation Contest. You can reach him via e-mail at anzovin@javanet.com.

FIGURE 11. The pivot of the jaw bone has been placed approximately where the real jaw hinge would be, based on Dennis's composite skull structure.
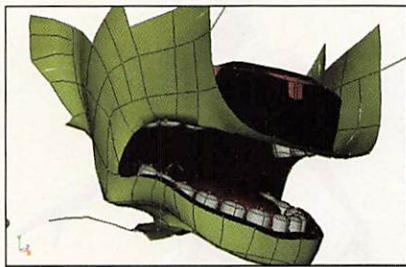

FIGURE 12. The jaw Smart Skin window. Rotating the jaw bone 15° causes translation and movement in the jaw area, bringing the mouth fully open.

sources

mirai

shape reality

ANIMATE YOURSELF.

Introducing Mirai 1.0

for Windows NT™, SGI™, and the SGI Visual Workstations™

Get ready for an animation package with so much creative freedom, you'll be moved to greatness. Get ready for Mirai 1.0. The next phase in the evolution of the popular N-World suite of tools, Mirai features a completely redesigned, remarkably intuitive interface. But even though it's the easiest system to use, it's also the most powerful and most integrated solution available.

At the heart of the system is the most advanced set of 3D animation tools available. Revolutionary new skeletal IK and integrated biomechanical motion editing work seamlessly with a non-linear interface that enables users to "build" animation sequences in layers, much like a 2D artist would build an image in a paint program.

What's more, Mirai seamlessly integrates a host of powerful features at a reasonable price. You get subdivision surface modeling, 2D and 3D paint, revolutionary skeletal modeling, advanced 3D inverse kinematics, non-linear, channel-based animation editing, biomechanical motion editing, photo-realistic rendering, object based particle systems, physical simulation, full ASCII data export with a conversion API, and a work flow that's going to set you in motion.

So start now. Move yourself to our website at www.nichimen.com or call us at 310.577.0500.

nichimen

GRAPHICS

Rapid 3D #18

www.nichimen.com

Peter and Ori Ratner
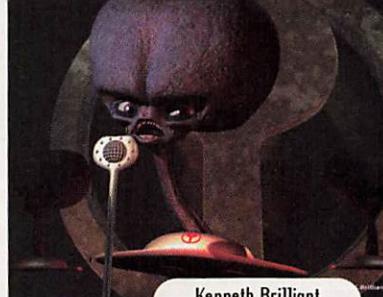
Kenneth Brilliant
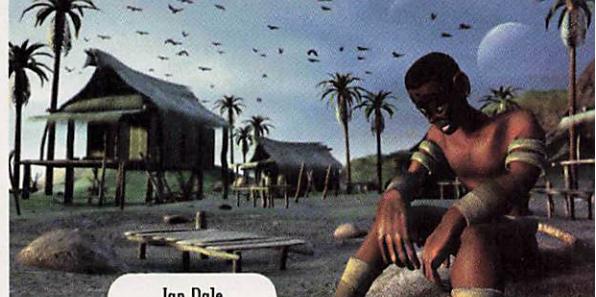
Ian Dale

1999 BIG KAHUNA NOMINEES

Once again, the search for the Big Kahuna brought piles of art and animation to the offices of *3D Design*. Our third annual Big Kahuna contest received hundreds of entries from all over the globe, and as we judged, we were amazed.

We present here the list of nominees in each category. Winners of each category—and the overall winner, the Big Kahuna—will be announced at the Big Kahuna Awards Ceremony at the 3D Design & Animation Conference, May 12, 1999, at the Santa Clara Convention Center in Santa Clara, CA. For more information, see the conference's official web site at www.3dshow.com. Congratulations to all the nominees!

3D Cartoon

Rod Bland
Not Without My Chicken X

Mitch Butler
The Smell of Horror

Waltuir and Guilherme Lopes
No Meio Do Caminho
Tinhauma Pedra,
Tinhauma Pedra No
Meio Do Caminho

Peter and Ori Ratner
A Bull's Tale

Print Graphics

Bryan Ballinger
Dodo Redux

Epic Studios
Spider

Kim Oravecz
You Wanna Iguana?

Interactive 3D

Peter Grundy
Best Intentions

Mike Malon
Caesar

Christopher Raphael
The Smithsonian Collection

Architectural Visualization

Glen Adomovicz
Titanic's Grand Staircase

Erich Aschenbrenner
Renovated Barn Kitchen

Tomasz Biernacki
Y2K Hideaway

William Munns
Emeisham At Night

Emmanuel Shiu
Restaurant Interior

Best Logo/ Corporate ID

John P. Chismar, Francine Izzo, and Victor Newman
NBC News Decision '98

Jeff Kasunic
Midnight Syndicate

Brian Osesek
Milwaukee Drill

Industrial Mechanical

Chad Curry
Easy Street

Igor Lodeiro
Central Air

Daniel J. Schwartz
DANA Frame Assembly

Fantasy World

Ian Dale
Left Behind

Ed Lee
Portal

David Michalczyk
Suburbs 2100

Eni Oken
Funhouse Variation

Non-Commercial Animation

Kenneth Brilliant
Safe Act

Dan Lemmon and Matthew Paulson
The Brief Social Life of Charles Wickham

Van Phan
Wild Card

Character Design

Kenneth Brilliant
Cell 107

Kenneth Brilliant
The General

**Cory Kruitbosch**
Enfish

Cory Kruitbosch
Ernst 3D

3D Film & Video Compositing

A special award will be presented in this category.
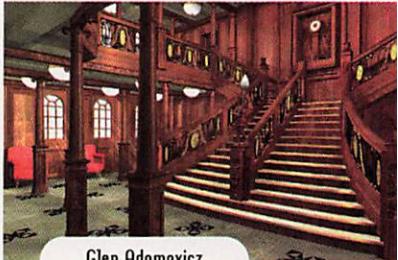

Glen Adomovicz
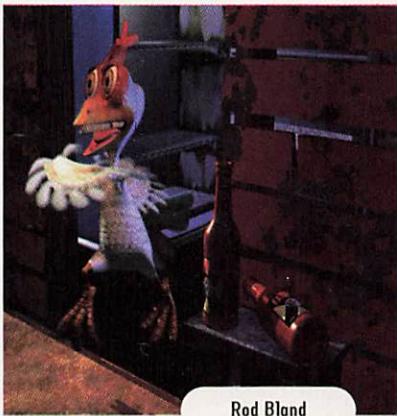
Rod Bland

Van Phan

Tomasz Biernacki

Dan Lemmon & Matthew Paulson

Life Forms Studio 3

Ever since the first appearance of NewTek LightWave for the Commodore Amiga, animators have searched for faster, easier methods to pose and animate CG characters. Six operating

systems and 5.6 versions later, LightWave has matured considerably and sports several features that aid in the struggle to bring digital characters to life. However, many artists still feel compelled to consider third-party tools for character animation. Joining the ranks of LightWave add-ons for this purpose is Life Forms from Credo Interactive.

The software comes in two versions, Life Forms 3 ($299) and Life Forms Studio 3 ($495). Both versions are available for Macintosh and Win95/NT 4.0 and are essentially identical, the difference being file format support. Life Forms 3 supports import/export for Caligari trueSpace4, Strata StudioPro, MetaCreations Infini-D 3, Kinetix 3D Studio R4, and Macromedia Extreme3D. Life Forms Studio 3 adds support for LightWave, 3D Studio MAX, BioVision BVH mocap files, Acclaim mocap files, and VRML 97, plus the PowerMoves collection of keyframed animations and motion capture files. For me, the main attraction is Life Forms Studio 3's ability to load and save LightWave objects and scene files complete with animation.

Coming To Life Installing Life Forms Studio 3 went smoothly under Windows 95. The QuickTime portion of the installation failed, but that was no surprise, since the documentation warned that it might happen. Running the supplied QuickTime Installer by itself completed the installation.
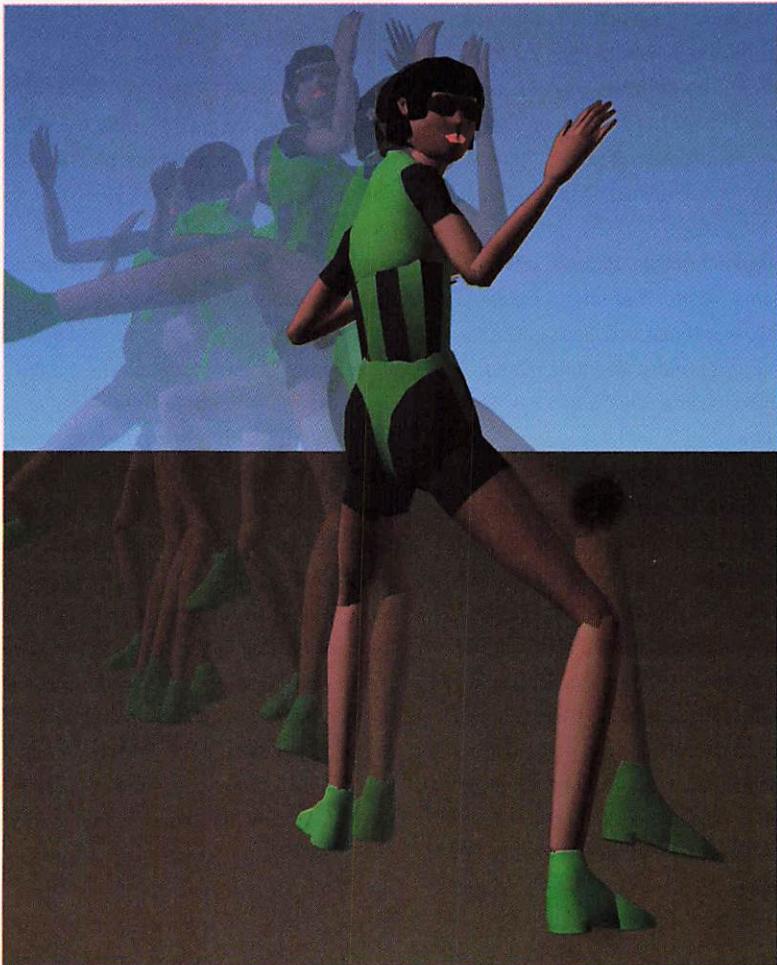
Credo provides quite a bit of introductory text to familiarize you with the user interface and relevant concepts before you get to the Quick Start tutorial, and ample reference documentation follows. In addition to the main manual, Life Forms ships with a reference for using Life Forms in conjunction with other programs. LightWave compatibility information isn't included in this book, but you can download it from Credo's web site.

The user interface is divided into various windows (Figure 1, p. 46). The Control Panel offers playback controls. In the Stage window, you place one or more figures and choreograph their position and orientation.

Faster, easier character animation using Credo Interactive's add-on for LightWave 3D, Life Forms Studio 3.

INTHELAB

In the Figure Editor, you pose them. The Timeline window lets you place keys on the desired frames, where they're displayed as thumbnails of the posed character—a wonderful feature. Most 3D animation apps display keyframes as small dots that look identical, making it hard to keep track of which key represents what pose. Thumbnails, on the other hand, allow for quick and easy identification of poses. You can get a sense of an animation just by looking at the time line.

In addition to these windows, you may also open a palette. A palette is a great way to store common or difficult poses for reuse. You simply drag keyframe thumbnails between the palette window and the Timeline window.

Finally, the Render window allows you to view a shaded representation of the figures on the Stage. The version I tested provided real-time interaction (limited to camera control) using QuickDraw 3D as well as a way to output animations to hard disk. OpenGL wasn't an option in the Render window, and my Diamond FireGL 1000 Pro seemed to have no effect on QuickDraw performance. (See "Coming Soon," p.47, for features of version 3.5.) Credo includes a demo for two plug-in renderers from LightWork (www.lightwork.com). One provides better photorealistic quality, and the other yields a black-and-white outline. The demos put a watermark over your renders, but you can activate the full versions for $89 each by contacting Credo.

When you launch the program, you're presented with an empty Stage and a fresh Timeline window. You can use the default models—male, male classic (which doesn't render as well), or female—or you can import a character hierarchy from one of the supported programs. The default models, which come with rotation constraints and a default pose (a basic standing pose with arms at sides) work particularly well with Life Forms. Character setup can take time to configure from scratch, so the animation-ready models are handy. You can start creating poses immediately, or you can load animations or
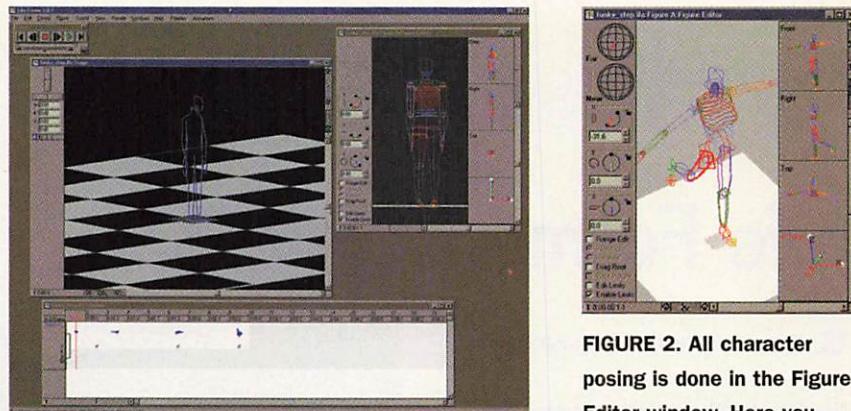
FIGURE 1. The main Life Forms interface windows are the Stage, the Figure Editor, the Timeline window, and the Control Panel.

FIGURE 2. All character posing is done in the Figure Editor window. Here you can see many of the tools available for creating poses.

poses from the supplied library of keyframed and motion-captured files.

The basic workflow consists of placing a figure in the Stage window, posing the figure in the Figure Editor to create keys, managing the keys in the Timeline window, then rendering or exporting the animation as geometry or bones with keyframes for use in another application. As soon as you change the default pose in the Figure Editor, it appears as a keyframe complete with thumbnail for the current frame in the Timeline (if a keyframe wasn't already there). You can then copy, cut, paste, or drag a keyframe anywhere in the Timeline or add it to a palette for later use. Another noteworthy feature is the handy animation browser, which allows you to preview animations before loading them into the Stage. This can be a tremendous help, especially in a multi-user environment where time is often wasted attempting to decipher the cryptic naming conventions of previous animators on a project.

Credo has taken a refreshing and somewhat unusual approach to simplifying the arduous task of creating and editing character animation, a process known as full hierarchy keying. Instead of handling keyframes on an object-by-object basis, Life Forms requires you to create keys for the entire hierarchy. Though some animators may find this method restrictive, it's closer to a traditional animation methodology than isolating keys and motion on an item-by-item basis, as is the case in most 3D animation packages.

Credo deserves praise for its effort to simplify the often complex and frustrating

process of hierarchy manipulation. Although the posing toolset has some notable oversights, I found myself creating natural poses with relative ease. My biggest complaint regards the amount of screen real estate devoted to the Figure Editor, which is the heart of the program. A large editing window is provided for selecting and rotating figure joints. Three small windows flank this window. You're presented with Front, Right, Top and Other views by default. These views are swapped among the three windows whenever you select one. Other viewport configurations would be welcome here.

Another cumbersome aspect of working in the Figure Editor is the fact that all actions are executed via the left mouse button only. Accommodating two or three mouse buttons (or key modifiers such as, Alt, or Option for single-button mouse systems such as Macintosh) would make posing a character in Life Forms even easier, particularly for joint translation (as opposed to rotation). It would be much more convenient to be able to modify the X, Y, and Z axes via left, middle, and right mouse buttons, respectively. For systems with only a single mouse button, a choice between mouse button, mouse button with Ctrl/Option, and mouse button with Alt would make work go faster.

Nonetheless, Credo has incorporated handy tools for rotating hierarchy segments, or joints as they're referred to in Life Forms. In addition to the usual X, Y, Z numeric input slider or spinner, the program offers two hemisphere images labeled Near and Far. A line representing the selected joint's

current orientation is displayed in one of the hemispheres, near if the joint is facing toward the camera and far if it's facing away. You can adjust any of these rotation controls or click and drag directly on the joint, which updates the relevant values in real time as you drag. Life Forms also offers a coord-sys (the little Cartesian coordinate system icon found in many 3D modeling and animation programs) style representation of the joint axis in the lower right corner of the user interface, which updates as well (Figure 2). In addition, Life Forms offers clearly defined and easily edited joint rotation limits. When they're enabled, the joint rotation limits affect forward kinematics as well as inverse kinematics (IK).

You can switch between forward kinematics and IK purely based on the current

Coming Soon

By the time you read this, Credo expects to have released the next revision of Life Forms and Life Forms Studio, version 3.5. The following new features are scheduled:

- **Snap!** This feature will snap joint objects to a defined point in 3D space over time. Credo suggests that it will help eliminate sliding feet.
- **OpenGL support.** With OpenGL hardware, the new version will provide faster, higher quality rendering from within Life Forms.
- **Additional file format support.** The update will increase Life Forms Studio 3's file format support to encompass NewTek LightWave, Kinetix 3D Studio MAX, Alias|Wavefront Maya and PowerAnimator, Play ElectricImage, Maxon Cinema 4D XL, Strata StudioPro, Caligari trueSpace, and MetaCreations Poser 3.
- **Joint Map editor.** The new Joint Map editor will make it easier to map motion data among models within Life Forms.
- **Enhanced Render window.** Version 3.5 will include the ability to reposition figures from the Render window.
- **Enhanced file management** (Windows only). File navigation and previewing functions will be improved, especially over local networks.
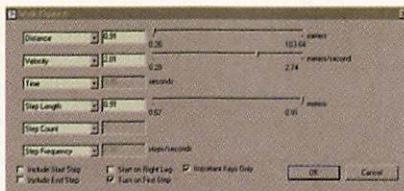
FIGURE 3. Although the walk generator isn't very robust, it's easy to use.

selection. If you select a single joint and drag with the mouse button depressed, the selected joint and its descendants will rotate. To utilize IK, you simply select two or more joints in a chain. Once the joints are selected, you can click and drag any of the selected joints and the entire selection adjusts according to IK.

The program's IK is useful, though it's not terribly sophisticated. Rotation constraints are crucial because the IK isn't goal-oriented. In a goal-oriented IK scenario, joint rotation limits or constraints aren't so critical because you can specify a different goal for each joint to follow, allowing for greater control during posing. With IK that isn't goal-oriented, you affect the chain by pulling one joint at a time, and thus the only way to achieve predictable results is to restrict how far each joint can rotate and on which axes. Although Life Forms lets you manipulate any joint in a selection, I usually achieved the best results by editing the end of the chain. When I dragged around other portions of the chain, things started to rotate in unpredictable ways and took time to reorient.

A feature called Drag Root in the Figure Editor allows you to constrain a joint's position and orientation while one or more of its ancestors are manipulated. You simply select the joint or joints you wish to lock, turn on the Drag Root feature, hold down the Shift key to select an ancestor (grandparent or higher), then move the ancestor to the desired position. This is especially useful for locking a character's feet to the ground while moving the body around. If you set up the joint restrictions properly, the legs will bend to accommodate the new body position. This feature is useful, but since you have only one mouse button for controlling the X, Y, and Z translation of the ancestor, it's quite difficult to use.

A selection menu pops up in the Figure Editor if you click and drag in a blank area of
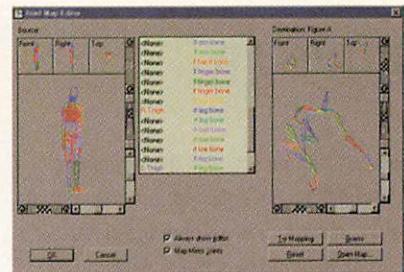
FIGURE 4. The Joint Map Editor allows you to determine exactly how motion will translate from one character hierarchy to the next on a joint-by-joint basis. This allows a great deal of flexibility when transferring animation from character to character.

the main window. It has some very useful options, such as Select All, Select Ancestors, and Select Descendants. Unfortunately, it's too easy to access this menu accidentally during failed attempts to click and drag on a joint. The menu selection also precludes bounding box or lasso selection, which would be more efficient in many cases. This is another place where multiple mouse-button support or key modifiers would be helpful.

Life Forms has a walk generator (Figure 3) for creating walk cycles or walking along a path. There's even an interface for re-mapping joints from the default figure to an imported hierarchy (Figure 4). The path animation is restricted to straight path segments and is best used to generate keys for further manipulation using the capabilities already described. Life Forms also allows you to synch audio to animations (the Mac version even allows you to use CD audio tracks). No further audio tools are provided, so you'll need to measure your audio timings elsewhere, then load a finished audio file into Life Forms.

I downloaded the LightWave import/ export plug-in from Credo's web site, and it worked well—at least initially (Figure 5). Life Forms is savvy about LightWave content directories, loading and saving objects to the appropriate folders. Although mesh deformation isn't supported directly in Life Forms, bone information can be manipulated and exported for use in LightWave. Unfortunately, IK limits and rotation limits are not exchanged between Life Forms and LightWave. Also, Life Forms exports the base of the character spine as a

whole torso object, and the remaining torso pieces have flipped surface normals, which makes them appear inside-out. This can be rectified by setting the first segment's Object Dissolve value to 100 percent, then loading the small body pieces into Modeler, flipping the normals, and resaving them.

No program is perfect, and Life Forms has its share of quirks. The Stage navigation tools are unusual and a bit awkward (at least compared to popular PC products such as LightWave, 3D Studio MAX, and Hash Animation:Master). Life Forms provides sliders and arrow buttons along each edit window's frame for camera control. These controls are unlike any I've come across elsewhere. One particular annoyance is that the horizontal and vertical scroll bars are not view-specific; if your fig-
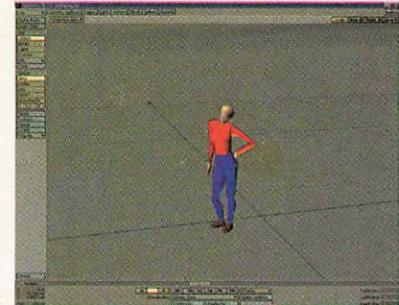
FIGURE 5. It was relatively simple to make this character "give attitude" with Life Forms 3. However, when imported into LightWave, the character's back segments were replaced with a series of nulls (some of them are visible just below the head) and a single torso object.

ure is off center, you'll probably need to scroll to achieve the proper camera position each time you switch views.

Speaking of views, Life Forms uses a single 3D editing window scheme in the Stage window and an asymmetrical quad view (one big window with three smaller windows to the right, as I mentioned before) in the Figure Editor. A standard quad view would be a welcome addition to both. Although you can choose between Linear and Spline figure shape interpolation settings, I couldn't see a difference. Another minor annoyance is the Frame Rate dialog box. It offers three input fields, but only two accept input. The manual states that you can change the frame rate by adjusting either the Frames Per Second field or the Length In Seconds field. The two fields I was able to edit were Frames Per Minute and Length In Seconds. Any value typed into the Frames Per Second field was ignored.

For all the gains provided by Life Forms Studio 3's simple setup and execution, export issues seriously threaten the product's usefulness. The import and export of LightWave scenes and geometry seemed almost flawless at first. Upon closer inspection, it became clear that something wasn't quite right. This boiled down to two minor annoyances. The first stemmed from the fact that Life Forms doesn't maintain separate paths for its Load and Export dialogs. Since I was loading animations from the Life Forms sample directory and exporting to a separate LightWave content directory, I had to change the path each time I

sources

switched between loading and exporting. This doesn't seem like such a big deal—that is, not until you've done it four or five times. The second annoyance was due to the fact that scenes exported from Life Forms load into LightWave with all objects set to display as bounding boxes. I prefer Full Solid display mode and had to change the setting for each scene I exported.

These problems paled in comparison with two further discoveries. With the exception of the supplied motion capture scenes, rotation key errors occurred on certain joints in the character hierarchies. This caused limbs to flail unexpectedly during playback in LightWave. The most common offenders were the lower legs, or knee joints as Life Forms calls them. These errors appeared only in scenes that didn't have keys on every frame. Furthermore, I found that bone Rest Direction information wasn't handled properly, causing strange deformations. If you create a scene with bones in LightWave and load it into Life Forms and export it, all the bones in the scene will have their Rest Direction reset to 0,0,0. If you create a scene from scratch in Life Forms and then export it to LightWave, the Rest Direction values for each bone seem to be random numbers.

Life Forms Studio 3 may be just what the doctor ordered for animators who have grown weary of complex character setup in LightWave. Unfortunately, Credo needs to do some more work under the hood before Life Forms can be considered a true contender in the LightWave animation race. In addition to squashing those motion bugs, Credo should take a look at LightWave, Maya, and Softimage for user interface cues. Although Life Forms' basic approach to 3D character posing and animation is definitely a step in the right direction, some of the implementation impedes the work-

flow. With the addition of a standard quad-view display, view/camera control, and multiple mouse-button support, Life Forms would be the ideal character animation tool to complement LightWave's modeling and rendering capabilities. If you're looking for decent mocap material to use with LightWave, Life Forms Studio 3 has it. Otherwise, you may want to wait for the next rev. ●

Paul Davies started his career as an illustrator. In 1989, Paul began creating digital images and animation for a small television post-production house in Virginia. Since that time, he has created CGI for print, multimedia, and games. Paul is currently doing modeling and texturing for games at Lucas Arts. Contact him at pdavies@lucasarts.com.

Cards for play.

These days, most consumer-level cards can handle 3D graphics. But try using one of those cards in 3D at 1280x1024, at a high refresh rate, and true color, and performance plummets dramatically. Even worse, features needed for professional applications like antialiasing, alpha blending, and stencil planes may not even be supported. And, while most of these cards claim OpenGL readiness, they haven't been tested with key applications. If you're a professional using 3D CAD or animation applications, you need a card that means business.

E&S has the solution–a new card specifically geared for professional users–*E&S Lightning 1200.*™

At $699, it's easy on your wallet so you don't have to compromise on a low-end mainstream card. It features 31MB of smart memory (16MB dedicated to textures) and provides true-color 3D resolutions to 1280x1024. With 3.3 million triangles per second and 70 million textured pixels per second, *E&S Lightning 1200* delivers the performance you need.

E&S Lightning 1200 is designed for and tested with professional applications like SolidWorks, Solid Edge, AutoCAD 2000, Bentley MicroStation, I-DEAS Artisan, Lightwave 3D, 3D Studio MAX, and for price-sensitive users of high-end applications like Pro/ENGINEER, I-DEAS Master Series, Unigraphics, SOFTIMAGE, and Maya.

E&S Lightning 1200, based on REALimage® technology, features DYNAMICgeometry™ drivers, which are the result of 24 months of joint effort with Intel. By fully utilizing the new Streaming SIMD Extensions, you get the power of a geometry accelerator in your CPU. It's no surprise that Intel chose E&S graphics cards to demonstrate their new Pentium®III processor.

If you'd like a more detailed comparison of consumer cards and E&S professional cards, send us an email at wginfo@es.com, or visit our web site at www.es.com.

Don't play games with your business. Choose a professional card designed for the work you do.

Cards for work.

E&S Lightning 1200™

Rapid 3D #21

Continuing Character Setup in Softimage|3D

◼ **To be transformed into a convincing animated character, a model must be prepared for all the kinds of motion you intend to create. This preparation—basically everything between modeling and animating—is called character setup.**

In the first part of this series on character setup in Softimage, we looked at constructing a character's skeleton (see "Character Setup in Softimage," May 1999, p. 49). A web-based supplement discussed attaching the character to the skeleton using Softimage's enveloping process (visit "Web Only Content" at www.3d-design.com). If you've followed along this far, your character should be able to bend and move with its skeleton. In this installment, we'll make a set of animation controls so the character will be easy to manipulate.

As I mentioned in part one, lack of foresight during the setup phase will mean a limited performance during animation. Character setup should allow for a complete range of movements, poses, and emotions. Your ability to pose and animate a character quickly will be aided greatly through the use of constraints.

A constraint is simply a way to attach or relate one object to another object. Constraints free you from the traditional limitations of the hierarchy. Think of a simple skeleton, designed in a linear fashion from the torso out to the limbs. When you move or rotate a parent in the hierarchy, all the chil-

dren follow along. But what if you wanted the hands or feet to stay put or the head to sit still while the shoulders rotate? That's where constraints come in.

Softimage|3D certainly isn't the only program that allows for setups using constraints and expressions (a mathematical relationship between two objects similar to a constraint), but it has been the standard in character setup for years. Despite its slow devel-

Now that your Softimage|3D character has developed some backbone, the setup process continues with a look at controls, constraints, and expressions.

opment and the long wait for Sumatra (the next generation of Softimage|3D), it remains one of the strongest 3D programs available for creating animated characters. Softimage|3D allows you to take a freeform approach to character setups; it's an open workflow that doesn't feel like an add-on or a plug-in. Softimage|3D's toolset lets you build a working virtual machine for manipulating your character.

To demonstrate the power of constraints, we'll continue using Stuart, the star of part one, who was designed originally for an in-house project at my company, Cineframe Animation.

Without using constraints, you could certainly pose and keyframe the end effectors themselves. However, there's a good reason to use constraints instead. A keyframed end effector can't have a parent relationship—its position can relate only to global coordinates. If you were to need to move a character globally after animating it, any end effectors you had keyframed would drag behind, pointing toward the last keyframed position. You can avoid this problem by constraining end effectors to other control objects, which, unlike end effectors, can have parent relationships.

Using constraints, you can create a hierarchy of control objects separate from the skeleton that have their parent-child relationships. For example, some of these relationships might rotate the hips while the torso stays still, while others might rotate the hips and the torso together. The skeleton follows the constraint objects, which in turn drive deformations and movements in the body. This type of character setup is three-tiered: consisting of body geometry, skeleton, and constraints.

In Softimage|3D's Schematic view, the skeleton and body appear as one hierarchy and the constraints are connected with yellow lines to the individual constraint objects. The three-tiered system looks like a traditional marionette in the schematic window; the character animation controls (also called constraint objects) look like wood crossbars, and the constraint lines resemble mari-
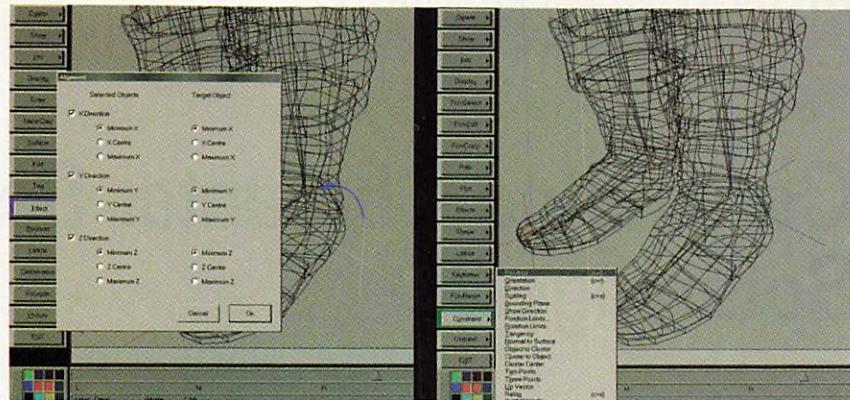
FIGURE 1. You can use Effect→Alignment to get the null into position, then use Constraint→Position to attach the leg end effector to the null (now colored blue and scaled bigger for easy selection).

onette strings. As a result, this system is sometimes referred to as the "marionette" approach.

In Softimage|3D, a constraint object isn't a special kind of object. It's simply any object to which another object (or objects) has been constrained. Normally, you'd choose something generic, like a null or a spline—a simple object that won't render and that's easy to see and select.

Leg Constraints Let's start with the legs. We'll use a null as a constraint object for each leg. To create the first null, choose Get→Primitive→Null. We need to place this null in the same position as the left leg end effector. With the null selected, choose Effect→Alignment. In the dialog, select the X, Y, and Z Direction check boxes. Click OK, then select one of the leg end effectors. The null jumps to the position of the end effector. To complete the process, select the end effector and choose Constraint→ Position, then select the null. Once the constraint is active, you can no longer move the end effector—its position is determined entirely by the position of the null (Figure 1). Depending on how large your character is in the workspace, you may want to scale the null up to make it easier to see (even in shaded mode with Show Icons active).

To control the plane of the 2D chain created in part one, you can apply an Up Vector constraint object. (This object controls the orientation of the chain, for
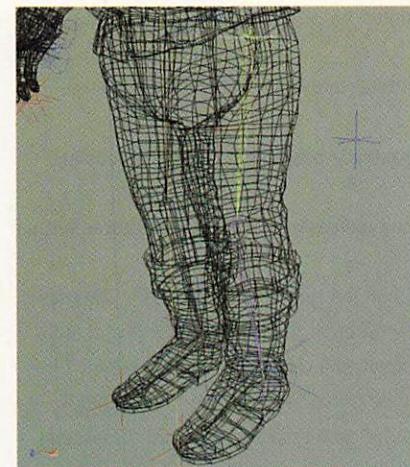
FIGURE 2. One way to control the plane of a 2D chain is an Up Vector constraint. Here, the first joint of the leg points along its Y axis to the null. Moving the null from side to side twists the leg chain, making the knee point in or out.

instance, the positioning of the knees and arms in relation to the body.) Position a null behind the body near the legs. It's best if you position it exactly on the current plane of the legs to maintain the default pose. To do so, open an ortho window with the upper leg joint selected. Choose Y directly above the window to align the view with the Y axis of the selected object. Position the null exactly behind the middle line in the joint of the upper leg (Figure 2). Select the joint again, choose Constraint→Up Vector, then select the null. The leg chain should point directly away from the null without moving from its cur-

rent orientation. Now, add the null into the constraint hierarchy. Repeat the process for the other leg, and finally parent both constraint nulls to a parent null.

Saving a translation keyframe (SaveKey→Object→Explicit Translation→All) on each of the leg nulls lets you experiment freely with the positions of the legs without losing the original positions of the nulls. Once the constraints are active, you can no longer reliably use Skeleton→Reset Actor to return the character to its default position; constrained parts will always snap back to their constrained positions after a refresh. If you place the constraints just right for the default position and save keyframes, you can always return to the original pose when you need to.

The constraint for the hips chain (with its root near the belly) should be the main parent of the rest of the body. Instead of using a null this time, it might be easier to use a spline (in this case a primitive square, which is a linear spline drawn in a square shape). Spline objects, like nulls, won't show up in renders and they aren't solid-shaded in OpenGL mode. They appear to hover over your character's geometry, so they're easier to select. And, unlike nulls, spline points can be scaled to whatever size is convenient, while the center remains at the default scale of 1.

Choose Get→Primitive→Square. Rotate 90° in X, then choose Effect→Freeze→ Transformations. To position the square at the hips, choose Constraint→Position (instead of Effect→Alignment) and select the root of the hips chain. We also want this control to rotate the hips, so choose Constraint→Orientation and then select the hips' root. The square moves and rotates into position. Next, you need to undo the constraints while keeping them in place. Choose Constraint→Relax and hit the middle mouse button to relax both constraints. Now repeat the process the other way, constraining the hips' root to the square's position and orientation. Select the root of the hips chain, choose Constraint→Position, and select the square.

Repeat this process with Constraint→Orientation, and parent your new hips control to the parent of the two leg controls. To keep the default pose (as I mentioned earlier), save an Explicit Translation and Rotation keyframe on the square. This control now rotates and translates the whole body, except the legs.

Independent Hip Rotation To control hip rotation independently of the upper body, you can simply rotate the hip joint, which is just beneath the root that was constrained earlier. Because the two leg chains are parented to this joint (or the end effector), the upper legs will always move with the hips. But selecting joints in shaded mode is problematic. Unfortunately, Softimage|3D doesn't have an X-ray mode that allows you to see joints within the shaded geometry, so grabbing the hip joint can be tricky if you aren't in wireframe mode. You could use the selectability feature to freeze the char-

acter's geometry and make it unselectable. To select the geometry you want to freeze, choose Select→Selectability→Toggle Selection. Still, without a wireframe window visible, it's hard to be sure what you've selected. Of course, you could select the joints in the schematic window. Instead, I find it simpler to keep applying constraint objects even to parts like this that don't *need* constraint objects but can benefit from the easier selection and visualization. This approach has the added benefit of maintaining all the animation information in one hierarchy and one animation file.

For areas like this, I tend to use a spline cube as a constraint object. (Again, any object can be used as a control.) Similar to a spline square, a spline cube is made from a single linear spline, drawn in the shape of the cube. In shaded view, the cube doesn't appear solid-shaded; only the lines show, hovering over the character's geometry. To make a spline cube, create a

Rapid 3D #22

default primitive cube, then use Magnet→On Point (located in the window's Layout Options menu) to draw a linear curve over each point and segment of the real cube. When you're finished, discard the default cube and your spline cube is ready.

Use Constraint→Orientation and Constraint→Position to move the spline cube into position over the hip joint. In this situation, you'll never need the constraint to translate the joint, only to rotate it, so after relaxing the translation constraints, use Constraint→Orientation on the hips joint to the spline cube. Note that constraints can be used to aid in forward kinematics as well as inverse kinematics (IK). In this situation, one constraint is used for driving rotations only (forward kinematics), while others affect translation (as with the legs) and still others affect both rotation and translation (as with the hips' root).

Parent the spline cube to the spline square hip control, then save a translation and rotation keyframe. You can also tag the points of the spline cube and translate them to the best position for visualization purposes. This is fine, since you're not changing the object's center. The spline cube will function the same way as before, but it will appear to hover over the entire area affected by it rather than over the root only (Figure 3).

Inverse and Forward Kinematics A useful arm setup will allow you to switch between inverse and forward kinematics. If a character is leaning on a desk, for example, you'll want the arm to stay positionally independent from the body. As you make changes in the spine and hips, the hand can stay locked to the desk through IK. But when the position of the hand isn't the top priority, many animators prefer the look and control of forward kinematics. It's easier to overlap the rotations of the arm joints— normally, you'd want the lower arm to rotate into place slightly after the upper arm. This procedure is difficult to simulate using IK.

The simplest way to allow for both inverse and forward kinematics is to use the
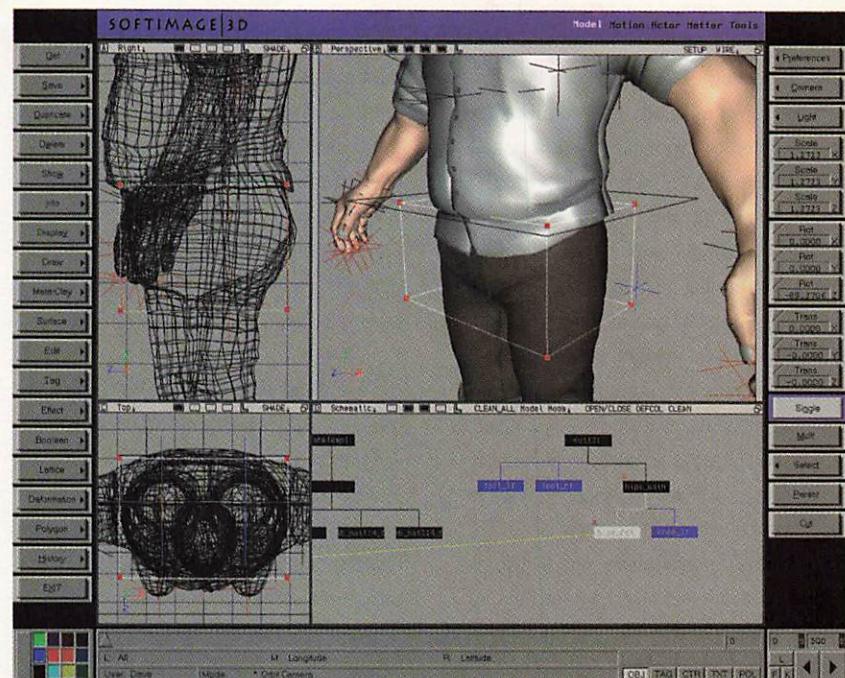
FIGURE 3. The points of a spline cube can hover conveniently over the hips area, while its center (the red and green arrows in the upper left window) remains in position at the root of the hips chain.

dopesheet to deactivate a portion of the time line for the relevant constraint. To do this, apply a constraint null to an arm the way you did earlier for the legs. Select the arm end effector and open the dopesheet. Click once to the left of the green bar. You should see "CnsPos" in the green bar, which indicates that a position constraint has been applied to the end effector. Drag a box selection over part of the time line. Click DEACT just above in the dopesheet window. That portion of the time line will turn gray, and the end effector will no longer be constrained to the null in those frames.

A more reliable and interactive way to allow for both inverse and forward kinematics is to apply an expression to the arm end effector telling it to attach to a null only if a certain condition is met. For instance, you can get a third null to act as a switch: When it translates to or above a value of 1 in Y, the IK constraints will become active. Otherwise, you can use direct forward kinematics on the joints.

To make this setup, get a null to act as the switch (called a slider), and name it ikswitch (in Info→Selection). Name the arm control arm_lf. Select the end effector and

choose Motion→Expressions→Edit. Type "etrnx" in the Affected Element portion after the object name or select etrnx from Fcurves. Etrnx is the function curve that describes the explicit translation in X.

For the part of the box labeled Expression, put your cursor in the field, then click on Condition in the Functions box. Click Insert, which inserts the condition "template" into the Expression area. Select the <cond> part and type over it with the condition part of the equation. Since you want the expression to apply only when the slider null is at or above a value of 1 in Y, type in ikswitch.etrny>=1, which means "if ikswitch is equal to or greater than 1 in Y translation..." Now replace the <true_expr> part with arm_lf.etrnx. This means "then make my value the same as arm_lf." For the <false_expr>, type in this.etrnx, which means "otherwise, let my value remain unchanged (do nothing)." Hit Validate to validate the script and make sure you have no syntax errors (Figure 4, p. 56).

Select the Expression part and click Copy. Click on Next, then paste it into the expression part. Change the Xs to Ys, including the one in the affected element.

FIGURE 4. An expression can be used to control whether an end effector will be constrained to a null at any given time. This expression tells the end effector to follow its null, arm_lf, only when the null ikswitch is greater than or equal to 1.

Hit Validate, then repeat the process for Z. This will give you three expressions, one for each axis.

Try out the expressions by translating ikswitch up to 1, then moving arm_lf around. The arm end effector follows. Move ikswitch to 0, and the arm will no longer be constrained to the null. For constraint switching during animation, simply animate the Y position of the ikswitch null.

If you attach an Up Vector constraint to the arm chain (as you did previously with the legs), you'll need to add expressions deactivating that constraint as well.

Alternatively, you might apply constraints to the joints for the forward kinematic motions, the way you did with the hips rotation control. Joint rotation controls are easier to select, and they maintain all the body animation under one null. However, they're more difficult to set up because you have to turn some constraints off and turn others on at the same time.

Whichever method you use, be sure to parent any constraint objects to the appropriate parent in the constraint hierarchy (in this case, either the main null or the main hips constraint). And don't forget to save an explicit translation and/or rotation keyframe.

Spine and Chest A common approach to constructing a spine is to make a three-joint
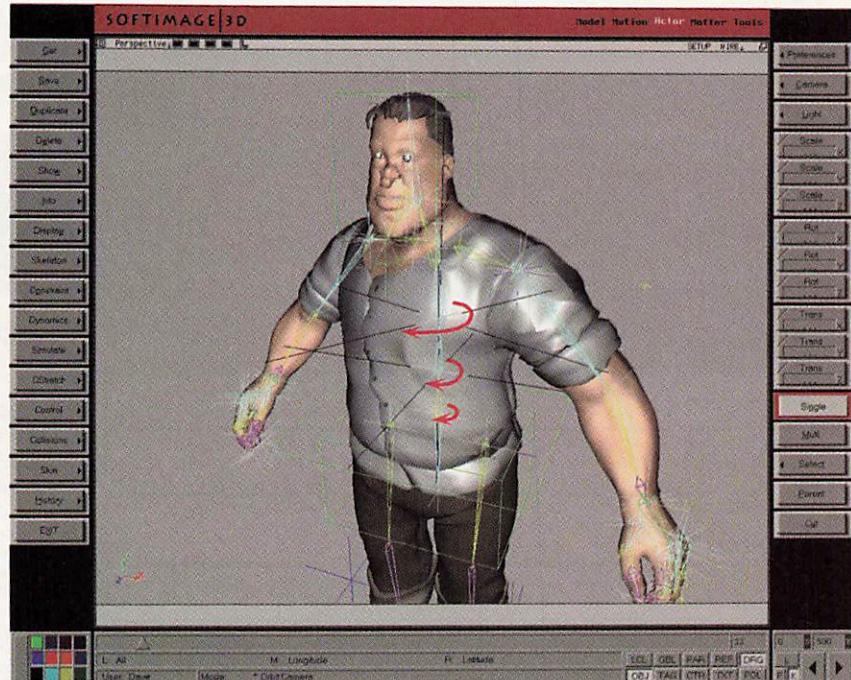
FIGURE 5. In this top-down approach to spine control, each node up the spine can be translated for positioning while the gradual twisting down the spine is controlled through expressions based on the rotation of the top (chest) joint. (Note: This screen shot is a composite. Softimage|3D doesn't have a display mode that allows a skeleton to show through shaded surfaces.)

chain from the hips to the base of the neck. Since every node in a spine must be able to twist in X as well as rotate in Y and Z, a 3D chain should be used. (Note: Drawing 3D chains in the Right window is different than with 2D chains—the Z axis always points along the global Z axis. You may want to draw a 2D chain and switch it to a 3D chain using Skeleton→Toggle 2D/3D for more consistent rotation values.) When animating, pose the spine and chest by beginning at the bottom of the chain and rotate each joint successively to the top.

An easier approach might be to control the motion from the top down using what could be called an artificial chain. That is, instead of building a multi-joint chain, you'll link together several single-joint chains. In the Right window, draw three single-joint chains up the spine, aligning the bottom joint's end effector with the next joint's chain root. Constrain each end effector to the next chain's root, up the line from bottom to top. Position several nulls between each chain, and constrain

each chain root to a null.

For the top chain, constrain the root both positionally (Constraint→Position) and orientationally (Constraint→Orientation). Now you can control the top of the spine via one null and each link using the other nulls. The top chain or joint can be positioned and rotated quickly and the other links translated quickly to form a pose. Using this method, the joints detach when the nulls are pulled out of reach, creating a stretching effect useful for spines. Note that your body's spine can stretch a bit as well. But the biggest advantage here is the ability to do top-down posing.

While each node can translate and the top node can translate and rotate in any direction, all twisting in X down the chain is controlled by the top node. Since the joints are not a single IK chain but a series of distinct chains linked artificially, you can drive their rotations through expressions. Otherwise, the expressions would conflict with the chain's built-in IK.

Because the top joint is controlled by a

null oriented like the chain root, you need to link the X rotation (twist) of each lower joint to the Y rotation of this chest null (roots are oriented differently than joints). Select the first joint below the top one and choose Expressions→Edit. Type "rotx" where the cursor is, in Affected Element. In Expression, type or select the name of the top null (from the Scn Elements box). After the name and period, type "roty" and then "/2" or 3 or 4, depending on how many joints you've made to represent the spine. This procedure tells the joint to rotate half as much in X as the chest null rotates in Y. Hit OK and repeat this process with each spine joint (you can use Expressions→Copy), decreasing the value of the rotation by increasing the division number. Last, you need to zero the rotation values of the chest null to make it work properly—make the value 0,0,0 with an effective value of 0,90,0. To do this, parent the chest null to a new null with a rotation of 0,90,0 (or parent to the hips square which is oriented this way) and set the chest null to 0,0,0. When the controlling null rotates, it will do so in local space, yielding more predictable rotation values. Its rotation will now twist the spine in decreasing amounts from the top down (Figure 5).

Many other methods can be employed to set up spines and spine-like structures such as tails and long necks. You can constrain a series of joints to clusters along a spline and animate the spline. You can deform the spline to another spline with fewer points for even finer control. This approach allows you to control many joints via very few points. You can drive rotations of a series of joints by rotating a single null through expressions.

Remain Flexible Once you've set up constraints, it's easier to manipulate all of your controls using the Preferred Transformation feature. For instance, select the chest control and choose Info→Transformation Setup. Click on the Rotate button and choose Y (check off X and Z). Whenever you select the chest control, the transformation

mode will switch automatically to the preset choice. This feature is especially helpful for rotations, because you don't need to remember which axis you're most likely to use whenever you go to rotate the object. You can still use the other axes, of course, but this method can speed up the animation workflow quite a bit.

It's important to remember that a character setup can be changed as the need arises. Constraints can be deactivated and temporary constraints can be added. The whole thing can be thrown out for a particular shot and replaced by a setup that better fits the needs of the movement. What if your character has to lean its elbow on a counter, chin resting in hand, while moving its torso slightly and gesturing with the other hand? With the current setup, which is designed to accommodate hand positions only, this could be a very difficult shot. But if you design a setup to accommodate the specific motion you need, your anima-

tion will go much more smoothly.

We've looked at only a few options and essentials for character setups. With an engineer's attention and an eye open to the numerous possibilities, you'll find character setup a fascinating study and an amazing help during character animation. ●

David Gallagher runs Utah-based Cineframe Animation (www.cineframe.com), a studio specializing in 3D characters. He's the author of the *Softimage Character Kit* training video series and designed the Softimage QuickBlend and YouMap plugins. Contact him at dave@cineframe.com.

sources

DIRECT, INSTANT, FREE

Visit **3D DESIGN** magazine's new product information service.

Access hundreds of 3D product web sites from one location with the

click of a button. Go to the RAPID 3D web site at **www.3d-design.com**.

ADVERTISER index *RAPID 3D GUIDE*

The index on this page is provided as a service to readers. The publisher does not assume any liability for errors or omissions.

RAPID 3D @ www.3d-design.com

RAPID 3D

3D Design

@ www.3d-design.com

Editorial Resources *RAPID 3D GUIDE*

AMD-K6
Advanced Micro Devices Inc. (AMD)
408-732-2400
RAPID 3D NO. 163

form•Z v. 3.0
form•Z RenderZone
auto•des•sys Inc.
614-488-8838
RAPID 3D NO. 159

MCXpress
RAPID 3D NO. 164
Softimage 3.8
RAPID 3D NO. 162
Avid Technology
978-640-6789

Life Forms 3
Life Forms Studio 3
Credo Interactive Inc.
604-291-6717
RAPID 3D NO. 161

3D Studio MAX
Character Studio

paint*, effect*
edit*
Lightscape
Discreet
415-547-2000
RAPID 3D NO. 165

ArchiCAD
Graphisoft Inc.
415-703-9777
RAPID 3D NO. 166

Animation:Master
Hash Inc.
360-750-0042
RAPID 3D NO. 160

Pentium III Xeon
Cascade
Intel
408-765-8080
RAPID 3D NO. 167

LightWork Renderer plug-ins
LightWork Design
+44 114 266 8404
RAPID 3D NO. 168

LightWave 3D
NewTek Inc.
210-370-8000
RAPID 3D NO. 169

Dreamcast
Sega
415-701-6018
RAPID 3D NO. 170

Softimage 3.8
Softimage
514-845-1636
RAPID 3D NO. 162

PlayStation
Sony
650-655-8000
RAPID 3D NO. 171

GET A JOB! SPECIAL SUPPLEMENT · ADVERTISERindex *RAPID 3D GUIDE*

RAPID 3D CONTEST

Go to Rapid 3D for product information and enter to WIN trueSpace4.

CALIGARI CORPORATION
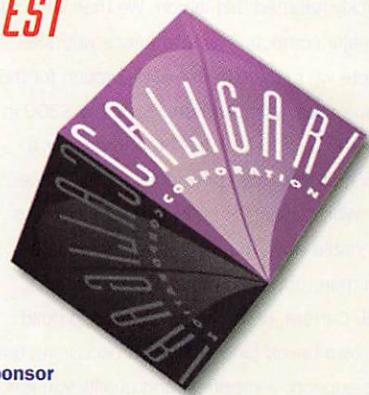
Rapid 3D Contest Prize Sponsor

BY CHRIS TOME

When a Clone is Not a Clone

● **The other day, I did something I've done a thousand times before. I helped a friend set up a new Windows PC clone workstation. She'd purchased it from a local company for her undergraduate animation work. She had heard good things from some other students about**

the company, so she took the plunge and spent more than $3,000 on a Pentium III beige box. Then the nightmare began.

The machine came in the usual made-in-Taiwan box, an unimpressive unit at best. Corners were cut in every possible way, and the components were cheap to say the least.

All right, I thought, it's lame, but no problem. It's here, it's a PIII, and I can still get the machine configured and the apps installed.

Not. After booting Windows 98 (she had bought both 98 and NT, but I knew not to let them install NT), I double-clicked on the My Computer icon, and that was all she wrote, so to speak. The computer blue-screened terminally, then the screen went black, and the box was completely hosed.

Not only was the PC built like a Yugo, it wasn't configured properly. It wasn't built to her specifications, and the support staff was one step above the help you get at a Taco Bell drive-thru window. Suffice it to say, we quickly returned this lemon. We then went to a major computer manufacturer's web site, where we found a better configuration for the same price. What's more, I took the $300 in tax she paid for the first PC and got her a box that was more powerful than the clone.

While clones or build-it-yourself used to be viable ways to save money, these days I'm more apt to go with a vendor like Dell, SGI, Carrera, or Intergraph. I used to build my own boxes (and still do on occasion), but the support, warranties, and quality you get in a box from a major vendor these days is worth the usually minimal premium.

That said, I would like to add that clones are not always a bad way to go. If

you aren't sure what to look for in a clone (or any other PC, for that matter), read on, because I've got a few pointers.

- A case should be large, somewhat stylish, and the power button shouldn't feel like it's going to snap off if you depress it too hard. If you have to install extra parts in your box, you'll appreciate a well-made case. Get at least a 300W power supply if you can. I've loaded too many cards into too many clones where the power wasn't sufficient and the box simply wouldn't function.
- Get as much RAM in as few sticks as possible. The manufacturer would prefer to get smaller, less costly DIMMS and fill up all of your slots. That may not seem bad now, but when you have to upgrade your RAM and find you have no extra slots, you'll be selling the extras you wind up with for pennies on the dollar. Slots average four per motherboard these days, and four 64MB sticks are cheaper than two 128s initially, but not in the long run.
- Get enough expansion slots! Face it, desktop form-factor computers are for Excel geeks. You need a minimum of a minitower case to accommodate a larger motherboard and all the hardware you'll eventually have, and you don't want to be stuck with a computer you can't add on to.
- A clone manufacturer should turn the machine on and burn it in for 72 hours before it ships. If, by that time, nothing is wrong with the computer, it should last a very long time, barring acts of God.
- Spec the parts yourself, and use only brand-name components for the big stuff.

You may pay more for brand-name parts, but you'll be happier in the long run.
- Get SCSI and forget about IDE. These days, the premium for getting SCSI, even at the low end, is so small that you should get the faster and newer of the two. IDE is definitely on its way out in the professional graphics space.
- Compare prices of components and get the best deals wherever you can. If, for example, you can find the graphics card you want for $100 less than a clone maker offers it for, get the box without a card, but have it configured for that card. It's pretty easy to install the card and get it working on your own.

There's much more to getting a good PC, but these points are a great place to start.

Caveat emptor takes on a greater meaning when you're talking about complex machines, and just as you would want a mechanic to check out a used car, you should either thoroughly check out any computer you intend to buy, or get a knowledgeable friend to do it for you.

It's easy to get taken—we've all had it happen sometime in our lives. It doesn't mean you're stupid unless you let it happen twice. Do your research, ask a lot of questions, and make sure you get what you asked for. You'll end up with a 3D tool that lets your imagination take flight. ●

Chris Tome is technical editor for *3D Design*, and he's had his hands inside more PCs than Bill Gates. If your PC is running, go catch it and e-mail him at ctome@mfi.com.

Creative Juice.

Power to the creators.

Quench your thirst for animation and digital video power with the creative juice of a TDZ®2000 ViZual™ Workstation from Intergraph Computer Systems.

A virtual stimulant for the imagination, one taste of this Z tells you it's unlike any other Windows® NT workstation.

100% pure hardware optimization ensures unmatched animation and digital video performance, while Intel® Pentium® III or Pentium III Xeon™ processors zap your apps with a powerful jolt of energy.

So, go ahead, satisfy your creative urges. Get juiced and grab the big purple box marked with a Z!
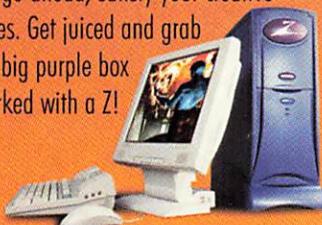
Want to know more about this sweet system? Call 1-800-763-0242 or visit us at www.intergraph.com/digitalmedia

INTERGRAPH
COMPUTER SYSTEMS

See Us at NAB 99 Booth S3948

Rapid 3D #39